

Adaptive Anchored Inversion for Gaussian Random Fields Using Nonlinear Data

Zepu Zhang

October 30, 2011

Abstract

In a broad and fundamental type of “inverse problems” in science, one infers a spatially distributed physical attribute based on observations of processes that are controlled by the spatial attribute in question. The data-generating field processes, known as “forward processes”, are usually nonlinear with respect to the spatial attribute, and are often defined non-analytically by a numerical model. The data often contain a large number of elements with significant inter-correlation. We propose a general statistical method to tackle this problem. The method is centered on a parameterization device called “anchors” and an iterative algorithm for deriving the distribution of anchors conditional on the observed data. The algorithm draws upon techniques of importance sampling and multivariate kernel density estimation with weighted samples. Anchors are selected automatically; the selection evolves in iterations in a way that is tailored to important features of the attribute field. The method and the algorithm are general with respect to the scientific nature and technical details of the forward processes. Conceptual and technical components render the method in contrast to standard approaches that are based on regularization or optimization. Some important features of the proposed method are demonstrated by examples from the earth sciences, including groundwater flow, rainfall-runoff, and seismic tomography.

Key words: adaptive model selection, dimension reduction, inverse problem, iterative algorithm, spatial statistics.

1 Introduction

Consider a spatial Gaussian process (also called a Gaussian random field) $Y(x)$, where $x \in \mathbb{D} \subset \mathbb{R}^3$, \mathbb{D} being the model domain, and $Y \in \mathbb{R}$. Define a *nonlinear* functional of this field: $\mathcal{M}: Y(\mathbb{D}) \rightarrow Z$, where $Z \in \mathbb{R}^{n_z}$ is a vector of length $n_z \geq 1$. Suppose we have observed the value of this functional, say the vector value is z . Conditional on this observation, how can we characterize the field $Y(x)$?

This is an abstraction of statistical approaches to a wide range of scientific questions. In these questions, $Y(x)$ is a “spatially-distributed” physical attribute. The functional \mathcal{M} , known as the “forward model” or “forward process”, is typically embodied in a deterministic, numerical algorithm that takes the field $Y(\mathbb{D})$ as input and outputs vector-valued result Z . In nature, the forward model is a physical process whose outcome is dictated by the attribute

zepu.zhang@gmail.com

Department of Mathematics and Statistics, University of Alaska, Fairbanks, Alaska, USA.

Published as *Inverse Problems* **27**(2011) 125011. doi:10.1088/0266-5611/27/12/125011

Y in the entire domain \mathbb{D} . The task is, conditional on the observed outcome of the forward model, how can we characterize (or “back out”) the attribute field Y ? Since we are using the outcome of the forward model to infer the quantity Y that dictates the outcome of the forward model, this task is known as an “inverse problem”.

A few examples will make the setting clear.

Example 1. [Yeh \[1986\]](#) reviews techniques for the groundwater inverse problem, in which the spatial attribute of interest is hydraulic conductivity of the subsurface medium. Conductivity controls groundwater flow which is governed by well-understood partial differential equations. Scientists or engineers make observations of groundwater movement to get head (i.e. pressure) and flux at selected locations and times, and attempt to use these data to infer the spatially distributed conductivity.

Example 2. [Bube and Burridge \[1983\]](#) study a 1-D problem in exploration seismology. In this setting, an impulsive or vibrating load applied at the ground surface launches elastic waves into the earth’s interior. Part of the wave energy is reflected by the medium and reaches the ground surface, where it is monitored at many time instants. Wave propagation in the subsurface column is described by a hyperbolic equation system with mechanic attributes of the elastic medium, including density and Lamé constants, as input parameters. The task is to recover the mechanic attributes, ideally everywhere in the subsurface column, using surface measurements of wave propagation including pressure and particle velocity.

Example 3. [Newsam and Enting \[1988\]](#) deduce the spatial distribution of CO_2 sources (including sinks) on the surface of the globe, given time-series observations of surface concentrations of CO_2 at some 20 locations around the globe. The CO_2 source here is treated as a spatial attribute defined for every location on the ground surface. The surface CO_2 concentration around the globe is the result of re-distribution of the sources via a transport process, governed by a diffusion equation.

These examples have three components in common: a spatial attribute of interest (hydraulic conductivity of flow media; mechanic properties of elastic media; source/sink of CO_2), a forward process (groundwater flow; wave propagation; atmospheric transport), and observed outcomes of the forward process (head and flux; pressure and particle velocity; CO_2 concentration). The attribute of interest varies in space and is possibly highly heterogeneous; in some scientific literature it is called a “(spatially) distributed parameter” [[Kravaris and Seinfeld, 1985](#); [Beven, 1993](#); [Nakagiri, 1993](#)]. The forward process is represented by a numerical model, which takes the spatial attribute as the main input, along with boundary and initial conditions. Usually, the need for this inversion arises because the attribute $Y(x)$ is needed for modeling (or simulating, or predicting) some other process of interest.

In concept, the unknown, $Y(\mathbb{D})$, has infinite dimensions. In real-world applications, it is almost always defined on a finite grid that discretizes the model domain. In either case, the unknown $Y(\mathbb{D})$ has many more dimensions than the known Z . This makes the inverse problem a severely under-determined one. It has become an established strategy to formulate the inverse problem in a statistical way [[Kirsch, 1996](#); [Kaipio and Somersalo, 2005](#)], that is, take $Y(x)$ as a random field and try to obtain its *conditional distribution* given the observation z , i.e. $p(Y(\mathbb{D})|z)$. Typically, this conditional distribution is represented empirically by a large number of *realizations* (i.e. simulations) of the random field. In this context, Bayesian approaches and Markov chain Monte Carlo computations have been widely used (see section 2).

In this study, we propose a method for this inverse problem. The proposal centers on a parameterization device called “anchors”, hence the name of the method—“anchored inversion”. Let the anchor parameters be denoted by θ . With anchor parameterization, *we solve a reduced problem*, namely, $p(\theta | z)$ rather than $p(Y(\mathbb{D}) | z)$. The dimension of θ is much lower than that of $Y(\mathbb{D})$ (the latter could be infinite), and the dimension of θ is under the user’s control. This dimension reduction brings advantages in computation as well as conceptualization.

The proposed method is a fundamental departure from some existing approaches. In section 2, we identify some challenges faced by existing methods, which motivate the concept of “anchor”. Anchor parameterization is formally described in section 3. Section 4 presents an iterative algorithm that approximates the posterior distribution of the chosen anchor parameter (θ) by a normal mixture. In section 5 we describe a strategy for choosing anchor parameters automatically and adaptively as the iterative algorithm proceeds. This efficient strategy makes anchor parameters evolve in iterations, adapted to the available data, complexity of the Y field, and the computational effort so far invested. This anchor-selection procedure is integrated into the algorithm described in section 4.

Section 6 describes two extensions to the basic framework presented in earlier sections. The first extension, geostatistical parameterization, is expected to be used in most applications. The second extension is used whenever the type of data described there are available. In section 7, the method is applied to three scientific questions using synthetic data. Section 8 concludes with a summary and highlights.

We emphasize “nonlinear” in this study, because treatment of “linear” forward models is relatively mature. A mainstay of the approaches is the Kalman filter and variants [Welch and Bishop, 1995]. However, the vast majority of forward processes in scientific applications are nonlinear, as demonstrated in section 7. Linear data, if available, are accommodated as described in section 6.2.

Although Y is “point-referenced” in concept, in practice the model domain is discretized (or aggregated) into a numerical grid. This discretization is assumed throughout, and it enables the use of matrix notations instead of integrals. We shall use x and Y as generic symbols for location and the spatial variable, and use \mathbf{x} and \mathbf{Y} for the finite-length x and Y vectors corresponding to the numerical grid. Specific values of Y and \mathbf{Y} are denoted by y and \mathbf{y} , respectively. Throughout, the symbol $\phi(\dots)$ denotes a normal density or likelihood function. We use $p(\cdot)$ as a generic symbol for probability density function and $p(\cdot | \cdot)$ for conditional density. When we have a specific density, say obtained by approximation, we use a specific symbol, for example $f(\cdot)$.

2 Motivation and rationale for the concept of anchor

Discretization of the model domain, along with other sources, introduces model error in the simulation of the forward process, \mathcal{M} . In addition there may be measurement error in z . Suppose the model and measurement errors are additive, we can write

$$z = \mathcal{M}(\mathbf{y}) + \epsilon, \tag{1}$$

where z is the forward data, \mathcal{M} is the forward model (i.e. a functional of the discretized field \mathbf{y}), and ϵ is combined model and measurement error.

A typical Bayesian approach takes \mathbf{Y} as the parameter vector and seeks its posterior distribution given the observation z . Suppose the density of the error ϵ is known to be $f(\epsilon)$, then the likelihood function is

$$p(z|\mathbf{y}) = f(z - \mathcal{M}(\mathbf{y})). \quad (2)$$

Upon specification of a prior $\pi(\mathbf{y})$, the posterior is $p(\mathbf{y}|z) \propto \pi(\mathbf{y})p(z|\mathbf{y})$. This high-dimensional posterior distribution is usually represented by samples from it via Markov chain Monte Carlo (MCMC) or related methods. There exist a large number of studies along this line, e.g. [van Leeuwen and Evensen \[1996\]](#); [Lee et al. \[2002\]](#); [Sambridge and Mosegaard \[2002\]](#); [Kaipio and Somersalo \[2005\]](#); [Tarantola \[2005\]](#). If the error is multiplicative with known distribution, manipulations may be possible in simple cases; see [Kaipio and Somersalo \[2005, p. 58\]](#) for an example. The construct above is also the cornerstone for a large class of approaches using “regularization” and optimization [[Tikhonov and Arsenin, 1977](#); [Kravaris and Seinfeld, 1985](#); [Engl and Neubauer, 2000](#); [Tenorio, 2001](#); [Doherty, 2003](#)].

We recognize three difficulties in this approach, namely, (1) that the same \mathbf{Y} is both the model parameter and the input to the forward model creates conflicting requirements; (2) the formulation is centered at “error”, ϵ , ruling out situations where error is nonexistent or negligible; (3) distribution of the error ϵ , especially the model error, is hard to specify. These points are elaborated below. Additional comments can be found in [O’Sullivan \[1986\]](#).

- (1) By taking the entire model grid \mathbf{Y} as the parameter vector, the dimension of the parameter space is tied to the spatial resolution of the numerical implementation of the forward model \mathcal{M} . This creates conflicts between parameter identifiability and accuracy in the numerical forward model \mathcal{M} , which is the essential connection between the model parameter and data. One side of this paradox calls for a coarse grid (for better identification of the model parameter) whereas the other side prefers a fine grid (for more accurate simulation of the forward process).
- (2) The only randomness in the parameter-data link in this formulation arises from the model and measurement errors, ϵ , because the forward model \mathcal{M} is deterministic. While in reality error usually exists, in synthetic or theoretical studies it does not have to. Clearly, the Bayesian approach centered at (2) does not apply if error is nonexistent. On the other hand, even if the magnitude of the error is not quite negligible, one may elect to ignore errors (because, for example, knowledge about the error’s distribution is very limited and unreliable) and still have a meaningful inverse problem of inferring \mathbf{Y} from z . However, the formulation based on (1) is not applicable in this situation.
- (3) Reliable knowledge of the error distribution, $f(\epsilon)$, is often unavailable. A major complication here is model error. While one may have a decent knowledge of the *measurement error* based on instrument specifications, *model error* is much more elusive. There are numerous sources for model error, including spatial and temporal discretization, inaccurate boundary and initial conditions, relevant physics that are omitted from consideration, and so on. Another notable error stems from the so-called incommensurability [[Ginn and Cushman, 1990](#)], i.e. the fact that what is computed by the model and what is measured in the field are not exactly the same quantity. A common

cause of incommensurability is discrepancy in the spatiotemporal resolution (or scale). Taken as a random vector, the model error usually has inter-correlated components. To further complicate the matter, the distribution of this error vector may depend on the input field \mathbf{y} .

Sambridge and Mosegaard [2002] point out that it is especially difficult to know the statistics of all errors when “the theoretical predictions from a model involve approximations”. However, all the sources of model error listed above essentially result in approximations.

The fact that ϵ contains model error has been emphasized by Scales and Tenorio [2001]; Lee et al. [2002]; Higdon et al. [2003]. At this point it must be recognized that model error may not be smaller than measurement error, hence it is not feasible to assume that measurement error dominates model error and makes the latter negligible. Tarantola and Valette [1982] give a telling example: “In seismology, the theoretical error made by solving the forward travel time problem is often one order of magnitude larger than the experimental error of reading the arrival time on a seismogram.” Also see Schoups et al. [2010].

We present an alternative approach that has the potential to alleviate these difficulties. The first intuition comes from the difficulty (1) above. Realizing that it is largely hopeless to aim for a sharp resolution of, say, a 100×100 field \mathbf{Y} using a data vector z of length 100, we ask, “Instead of taking on the entire field \mathbf{Y} , what about aiming for a sharper resolution of, say, 50 ‘characteristic values’ of \mathbf{Y} using the data z , and then parameterizing the field with these 50 characteristic values?”

The answer is to reduce the model parameter vector and separate it from the numerical grid. Specifically, we take certain linear functionals of the field as such characteristic values and call them “anchors”, denoted by θ :

$$\theta = \mathbf{H}\mathbf{Y}, \quad (3)$$

where \mathbf{H} is a $n_\theta \times n_{\mathbf{y}}$ matrix of rank n_θ , satisfying $n_\theta < n_{\mathbf{y}}$. Subsequently, our model inference involves deriving $p(\theta|z)$, and parameterizing the field \mathbf{Y} by θ via $p(\mathbf{y}|\theta)$. In a sense, we have proposed to solve a different problem, a “reduced” one, than that tackled by the standard approach (which derives $p(\mathbf{y}|z)$). In the reduced problem, the anchors act as “middleware” between z and \mathbf{Y} ; the most important information in z about \mathbf{Y} is “transferred” into θ , in the form of $p(\theta|z)$, instead of into \mathbf{Y} directly. We shall use the name “anchored inversion” for the proposed method. The key elements of this method are the anchor definition represented by the matrix \mathbf{H} , the anchor parameterization for the field, i.e. $p(\mathbf{y}|\theta)$, and derivation of the posterior $p(\theta|z)$. These issues are discussed in the sections to come.

It is apparent that the anchor concept de-couples the dimension of the model parameter (which is now θ) from that of the numerical grid. The former is under our control, and as such can be vastly lower than the latter. Whereas the dimension of \mathbf{Y} may be dictated by physics, required resolution and accuracy of the forward model, as well as subsequent application needs for the inferred attribute field, the dimension of θ is determined mainly by statistical and computational concerns. The user has the freedom to seek a trade-off,

via the choice of anchors, between feasible derivation of $p(\theta | z)$ (calling for a shorter θ) and sufficient parameterization of the field by $p(\mathbf{y} | \theta)$ (calling for a longer θ).

This spirit of dimension reduction is shared by an active line of research using “convolutions” [Higdon, 2002; Higdon et al., 2003; Short et al., 2010]. The convolution-based methods differ from anchored inversion in many ways. For one thing, the former conducts computation in a MCMC framework, whereas the latter is “incompatible” with MCMC, as addressed next.

The second difficulty in the existing approach, i.e. errors must be present for the formulation to be applicable, is eliminated. In the existing approach, the parameter-data connection is

$$\mathbf{y} \xrightarrow{\mathcal{M}(\mathbf{y}), \epsilon} z.$$

In contrast, the connection in anchored inversion is

$$\theta \xrightarrow{p(\mathbf{y} | \theta)} \mathbf{y} \xrightarrow{\mathcal{M}(\mathbf{y}), \epsilon} z. \quad (4)$$

The randomness in the parameter-data connection now comes from two sources (or layers): the random field \mathbf{y} conditional on parameter θ , and the random error ϵ conditional on fixed \mathbf{y} . The algorithm of anchored inversion (see section 4) learns about the statistical relation between the model parameter θ and the data z (or Z , to be more accurate) by simulating the data-generation process. Note that the first layer of randomness is tractable by simulation because it is due to the *known* parameterization $p(\mathbf{y} | \theta)$. The second layer of randomness can be simulated if one has decent knowledge about the errors. The “knowledge” here does not have to be a closed-form distribution; it may be, for example, mechanism of the occurrence of errors in just enough detail such that the (stochastic) mechanism can be embedded in the forward simulation.

However, if the dimension of θ is much lower than that of \mathbf{y} , the first layer of randomness could dominate the second, making an accurate quantification of ϵ less critical to accurately simulating the statistical relation between θ and z . In such situations, inaccurate but helpful quantification of the errors may be used, or the errors may be ignored altogether. Moreover, anchored inversion allows the situation where the forward model and measurements are error free because, lacking the second layer of randomness, the parameter-data relation is still a statistical one due to the first layer of randomness. (In fact, in the case studies presented in section 7, we did not artificially add model or measurement errors to the synthetic data.)

This leads to the most fundamental distinction between anchored inversion and MCMC methods: the former does not make analytical quantification of the statistical relation between the model parameter and data, but rather simulates the relation. In the simulation, a likelihood function is not known (and not needed); consequently, a standard application of the Bayes theorem is not possible.

Related to the third difficulty in the existing approach, general forms of errors, be it additive, multiplicative, or arising in multiple stages of the forward process \mathcal{M} , may be accommodated in the numerical derivation of the posterior $p(\theta | z)$. This will be discussed in section 4.2. Section 4.2 also suggests that it may be possible to apply the proposed method to inverse problems where the forward model is non-deterministic.

3 Anchor parameterization

We model $Y(x)$ as a Gaussian process. In other words, $\mathbf{Y}(\mathbf{x})$ is a (high dimensional) normal random vector with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$:

$$p(\mathbf{y} | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \phi(\mathbf{y} | \boldsymbol{\mu}, \boldsymbol{\Sigma}), \quad (5)$$

where ϕ is the normal density function. With the anchor θ defined in (3), we have the joint distribution

$$\begin{bmatrix} \mathbf{Y} \\ \theta \end{bmatrix} \sim N \left(\begin{bmatrix} \boldsymbol{\mu} \\ \mathbf{H}\boldsymbol{\mu} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma} & \boldsymbol{\Sigma}\mathbf{H}^T \\ \mathbf{H}\boldsymbol{\Sigma} & \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T \end{bmatrix} \right).$$

Then \mathbf{Y} conditional on θ is normal:

$$p(\mathbf{y} | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \phi \left(\mathbf{y} \mid \boldsymbol{\mu} + \boldsymbol{\Sigma}\mathbf{H}^T (\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T)^{-1} (\theta - \mathbf{H}\boldsymbol{\mu}), \boldsymbol{\Sigma} - \boldsymbol{\Sigma}\mathbf{H}^T (\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T)^{-1} \mathbf{H}\boldsymbol{\Sigma} \right). \quad (6)$$

The power of the anchor parameterization lies in the basic property of the normal distribution that leads to the conditional distribution (6). Depending on the choice of anchors (i.e. the matrix \mathbf{H}), this parameterization introduces flexible, and analytically known, mean and covariance structures into the field vector \mathbf{Y} .

We shall speak of “sampling” the conditional distribution $p(\mathbf{y} | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$. In actual implementations, an explicit sampling of (6) is often replaced by a “conditional simulation” procedure that consists of two steps:

- (a) Sample \mathbf{y}^* from $\phi(\mathbf{y} | \boldsymbol{\mu}, \boldsymbol{\Sigma})$.
- (b) Update \mathbf{y}^* to $\mathbf{y}_* = \mathbf{y}^* + \boldsymbol{\Sigma}\mathbf{H}^T (\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T)^{-1} (\theta - \mathbf{H}\mathbf{y}^*)$.

It is easily verified that the resultant \mathbf{y}_* is indeed a random draw from (6).

A typical application of anchored inversion consists of three steps:

- (a) Define anchors θ , i.e. specify \mathbf{H} .
- (b) Derive the distribution of θ conditional on data z , that is, $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$.
- (c) Generate a sample of field realizations using the posterior $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ and the conditional $p(\mathbf{y} | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$.

We first present an iterative algorithm in section 4 for approximating $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, assuming \mathbf{H} has been specified. In section 5, we present a procedure that automatically and adaptively specifies \mathbf{H} in the iterations of the algorithm. Therefore the anchors (i.e. \mathbf{H}) are not fixed, but rather evolve in the iterations. The third step entails sampling by “composition” [Tanner, 1996, sec 3.3.2], that is, sampling θ_* from $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ followed by sampling \mathbf{y}_* from $p(\mathbf{y} | \theta_*, \boldsymbol{\mu}, \boldsymbol{\Sigma})$. In these samplings, the anchor specification and their posterior distribution are taken from the final iteration of the algorithm. Because $p(\mathbf{y} | \theta_*, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is normal, the computational cost of the sampling is low. Sampling $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is also easy because, as we shall see in section 4, the distribution is approximated by a normal mixture distribution.

In section 6.1, we will introduce a small number of geostatistical parameters to characterize the mean vector $\boldsymbol{\mu}$ and the covariance matrix $\boldsymbol{\Sigma}$ of the field. These geostatistical parameters are regarded as unknown and are inferred in the iterative algorithm along with the anchor parameters.

4 Iterative conditional mixture approximation to the posterior distribution

We now tackle the posterior of the model parameter θ given data z , the mean vector $\boldsymbol{\mu}$, and the covariance matrix $\boldsymbol{\Sigma}$ of the field vector \mathbf{Y} . Let $\pi(\theta)$ be a prior for θ . This is specified by

$$\pi(\theta) = p(\theta | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \phi(\theta | \mathbf{H}\boldsymbol{\mu}, \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T) \quad (7)$$

following (3) and (5). Noticing the parameter-data connection depicted in (4) and the generality of the forward model $\mathcal{M}(\mathbf{y})$, the usual route through the Bayes theorem, $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \propto \pi(\theta) p(z | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is not usable here because the likelihood function $p(z | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is unknown. In fact, it can be difficult to analyze the statistical relation between θ and Z because \mathcal{M} is typically defined by a complicated numerical code, involving many components and steps, including possibly ad hoc operations.

However, we can sample $p(z | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ following the “data-generating” mechanism $\theta \rightarrow \mathbf{y} \rightarrow z$. Starting with a particular θ_* , this mechanism involves sampling \mathbf{y}_* from $\phi(\mathbf{y} | \theta_*, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ and subsequently evaluating $z_* = \mathcal{M}(\mathbf{y}_*)$. If θ_* is a random sample from the prior $\pi(\theta)$, then (θ_*, z_*) is a random sample from the joint distribution of θ and Z , namely $p(\theta, z | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \pi(\theta) p(z | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$.

In general, suppose (θ, Z) (conditional on $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$) has a mixture distribution with density

$$p(\theta, z | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{i=1}^n w_i p_i(\theta, z),$$

where p_i is the density of the i th mixture component and the weights w_i sum to unit. Denote the marginals of θ and Z corresponding to $p_i(\theta, z)$ by $p_i(\theta)$ and $p_i(z)$, respectively, and write the conditionals $p_i(\theta | z) = p_i(\theta, z)/p_i(z)$ and $p_i(z | \theta) = p_i(\theta, z)/p_i(\theta)$. The density of θ conditional on z as well as $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ is then

$$p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{p(\theta, z)}{\int_{\Theta} p(\theta, z) d\theta} = \frac{\sum_i w_i p_i(\theta, z)}{\sum_j w_j \int_{\Theta} p_j(\theta, z) d\theta} = \frac{\sum_i w_i p_i(\theta, z)}{\sum_j w_j p_j(z)} = \sum_i v_i p_i(\theta | z) \quad (8)$$

where $v_i = w_i p_i(z) / \sum_j w_j p_j(z)$. This shows that the conditional (or posterior) $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is a mixture of the conditionals $p_i(\theta | z)$.

Take a pause here and think about the situation we are in. First, it appears possible to obtain random samples of (θ, Z) . Second, the joint density of (θ, Z) may be approximated by a mixture distribution. Third, if it is feasible to get the marginal $p_i(z)$ and conditional $p_i(\theta | z)$ from the mixture component $p_i(\theta, z)$, we are on a clear path to $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, the ultimate target.

Regarding the last point, $p_i(z)$ and $p_i(\theta | z)$ are readily available if $p_i(\theta, z)$ is normal. The second point suggests kernel density estimation. In view of the third point, Gaussian kernels should be used. As for the first point, if we do not have random samples of (θ, Z) , we could have “weighted” samples by the general importance sampling technique [Geweke, 1989; Owen and Zhou, 2000].

In fact the algorithm below will show that importance sampling is critical for building an *iterative* procedure. Because we are able to sample (θ, Z) as needed (via simulations), we

sample it in iterations, each time using a manageable sample size and an “improved” proposal distribution. After obtaining a reasonably large sample of (θ, Z) , we embark on a Gaussian kernel density estimation for the joint distribution $p(\theta, z | \boldsymbol{\mu}, \boldsymbol{\Sigma})$, with the real goal being an estimate of the posterior (or conditional) $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$. Over iterations, the proposal distribution improves not in terms of estimating $p(\theta, z | \boldsymbol{\mu}, \boldsymbol{\Sigma})$ but in terms of estimating $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$.

4.1 Algorithm

We begin with an initial approximation to the posterior $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, denoted by $f^{(0)}(\theta)$, which is taken to be a sufficiently diffuse normal distribution. The first iteration updates $f^{(0)}$ to $f^{(1)}$. In general, the k th iteration updates the approximation $f^{(k-1)}$ to $f^{(k)}$ as follows.

1. *Drawing a random sample of the parameters.*

Take n random samples, denoted by $\theta_1, \dots, \theta_n$, from $f^{(k-1)}$. For $i = 1, \dots, n$, compute the prior density $s_i = \pi(\theta_i)$ according to (7), and the “proposal density” $t_i = f^{(k-1)}(\theta_i)$; let $w_i = \frac{s_i/t_i}{\sum_{j=1}^n s_j/t_j}$.

2. *Evaluating the forward model.*

For $i = 1, \dots, n$, sample \mathbf{y}_i from $p(\mathbf{y} | \theta_i, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ according to the density (6), and evaluate $z_i = \mathcal{M}(\mathbf{y}_i)$. The sampling of \mathbf{y}_i is typically accomplished by a geostatistical simulation algorithm outlined in section 3, while evaluation of z_i usually entails running a numerical forward model with the simulated field \mathbf{y}_i as input.

3. *Reducing the dimension of the conditioning data by principal component analysis (PCA).*

Let $\mathbf{Z} = [z_1 | \dots | z_n]^T$, where z_1, \dots, z_n are column vectors of length n_z . By PCA, we find a $n_z \times m$ matrix \mathbf{A} , where $m \leq n_z$, such that the matrix $\mathbf{Z}\mathbf{A}$ “explains” specified proportions (e.g. 99%) of the variations in \mathbf{Z} . Subsequently, we transform each simulated z_i , $i = 1, \dots, n$, to $\mathbf{A}^T z_i$, and also transform the observation vector z similarly. This transformation reduces the dimension of Z from n_z to m . To keep the notation simple, we shall continue to use the symbol z for this transformed variable.

4. *Estimating the joint density of (parameter, data).*

Based on the *weighted* sample $\{(\theta_1, z_1), \dots, (\theta_n, z_n)\}$ with weights w_i , approximate the density function of the random vector (θ, Z) by a multivariate normal mixture:

$$p(\theta, z) \approx \sum_{i=1}^n w_i p_i(\theta, z) = \sum_{i=1}^n w_i \phi(\theta, z | \mathbf{m}_i, \mathbf{V}_i), \quad (9)$$

where \mathbf{m}_i and \mathbf{V}_i are the mean vector and covariance matrix of the i th mixture component.

We use a normal-kernel density estimator with two tuning parameters: one is bandwidth, which is a standard tuning parameter; the other is a localization parameter, which specifies how large a fraction of the entire sample is used to calculate the covariance matrix of each mixture component. These two tuning parameters are determined

simultaneously by a likelihood-based optimization procedure. Because this step is a completely modular component in the entire algorithm (and alternative normal-kernel density estimators may be used without any change to other parts of the algorithm), we refer the reader to Zhang [2012] for details.

5. Conditioning on the data.

For $i = 1, \dots, n$, partition \mathbf{V}_i as $\begin{bmatrix} \mathbf{V}_{\theta\theta i} & \mathbf{V}_{\theta zi} \\ \mathbf{V}_{z\theta i} & \mathbf{V}_{zzi} \end{bmatrix}$, where $\mathbf{V}_{\theta\theta i}$ is the covariance matrix of θ , \mathbf{V}_{zzi} is the covariance matrix of Z , and $\mathbf{V}_{\theta zi}$ as well as $\mathbf{V}_{z\theta i}$ are cross-covariance matrices. Similarly, decompose \mathbf{m}_i into $\mathbf{m}_{\theta i}$ and \mathbf{m}_{zi} , representing the θ part and Z part, respectively. Use the notations $p_i(\theta)$, $p_i(z)$, $p_i(\theta | z)$, and $p_i(z | \theta)$ as defined for relation (8). Then

$$p_i(z) = \phi(z | \mathbf{m}_{zi}, \mathbf{V}_{zzi}) \quad (10)$$

and

$$p_i(\theta | z) = \phi(\theta | \mathbf{m}'_{\theta i}, \mathbf{V}'_{\theta\theta i}), \quad (11)$$

where

$$\mathbf{m}'_{\theta i} = \mathbf{m}_{\theta i} + \mathbf{V}_{\theta zi} \mathbf{V}_{zzi}^{-1} (z - \mathbf{m}_{zi}), \quad \mathbf{V}'_{\theta\theta i} = \mathbf{V}_{\theta\theta i} - \mathbf{V}_{\theta zi} \mathbf{V}_{zzi}^{-1} \mathbf{V}_{z\theta i}. \quad (12)$$

Here z is the observed forward data. Substituting (10)-(11) into (8), we get the conditional density

$$p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \approx \sum_{i=1}^n v_i \phi(\theta | \mathbf{m}'_{\theta i}, \mathbf{V}'_{\theta\theta i}), \quad \text{where } v_i = \frac{w_i p_i(z)}{\sum_{j=1}^n w_j p_j(z)}. \quad (13)$$

The posterior approximation $f^{(k-1)}$ is now updated to $f^{(k)}(\theta) = p(\theta | z)$. This completes an iteration of the algorithm.

Note $f^{(k)}$ is a normal mixture, just like $f^{(k-1)}$, and is ready to be updated in the next iteration.

4.2 Accommodating model and measurement errors

Suppose the combined model and measurement error in z is additive, as represented in (1). Assume ϵ is independent of θ , \mathbf{Y} , and Z . Further assume the error is normal, $\epsilon \sim N(\mathbf{0}, \boldsymbol{\Sigma}_\epsilon)$. Then the \mathbf{V}_{zzi} in (10) and (12) should be replaced by $\mathbf{V}_{zzi} + \boldsymbol{\Sigma}_\epsilon$.

The algorithm is able to accommodate more general errors. For example, non-additive errors and errors that might occur in certain internal steps of the forward model \mathcal{M} can be built into the forward evaluation $z_i = \mathcal{M}(\mathbf{y}_i)$ of step 2 by introducing randomness according to a description of the errors. This way, multiple forms of errors can be accommodated. An overall, explicit formula for the errors as a whole is not required. The passage from θ to z in steps 1–2 embodies our best knowledge of the parameter-data connection: part in the anchor parameterization $p(\mathbf{y} | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, part in the forward model $\mathcal{M}(\mathbf{y})$, and part in errors in our implementation of \mathcal{M} and in the measurement of z .

Going one step further, it is possible to relax the requirement that the forward model \mathcal{M} be “deterministic”. If the forward model is stochastic [see an example in Besag et al., 1991],

the parameter-data connection, as depicted in (4), includes one more layer of randomness. As far as the algorithm is concerned, the situation is not different from inclusion of model errors.

4.3 Comments

The overall idea in this algorithm may be summarized as “simulate $p(\theta, Z)$, then condition on the observed z .”

The algorithm does not require one to be able to draw a random sample from the prior of θ . Instead, a convenient initial approximation $f^{(0)}$ starts the procedure. One only needs to be able to evaluate the prior density at any particular value of θ . This provides flexibilities in specifying both the prior $\pi(\theta)$ and the initial approximation $f^{(0)}$.

Some statistics of the distribution $f^{(k)}$ may be examined semi-analytically, taking advantage of its being a normal mixture. More often, one is interested in the unknown field \mathbf{Y} or a function thereof rather than in the parameter θ . These may be investigated based on a sample of \mathbf{Y} based on the posterior of θ , as explained in section 3.

Typically, by far the most expensive operation in this algorithm is evaluating the forward model $\mathcal{M}(\mathbf{y})$. This occurs once for each sampled value of θ . In contrast, sampling θ or generating field realizations are easy, because the distributions involved are normal or normal mixtures.

The versatility of normal mixture in approximating complex densities is well documented [Marron and Wand, 1992; West, 1993; McLachlan, 2000]. This approximation requires every component of (θ, Z) to be a continuous variable defined on $(-\infty, \infty)$, hence some transformations may be necessary in the definition of θ and Z . (See section 7 for examples.)

The dimension reduction achieved by principal component analysis in step 3 can be significant, for example with time series data. This facilitates the use of high-dimensional (i.e. large n_z) data without worrying (too much) about the correlation between the data components. The dimension reduction is also a stability feature. For example, components of Z that are almost constant in the simulations will not cause trouble.

The global structure of this algorithm bears similarities to that of West [1993], with important differences. While West [1993] focuses on situations where the likelihood function is known, the algorithm here completely forgoes the requirement of a (or the) likelihood function. This distinction has far-reaching implications for the applicability of the proposed method.

This is an important feature of the proposed algorithm: it does not require a likelihood function. This feature distinguishes itself from the Markov chain Monte Carlo (MCMC) methods. There is a variant of MCMC called “approximate Bayesian computation (ABC)” [see Marjoram et al., 2003] that applies MCMC in problems without known likelihood functions. ABC uses a measure of the simulation-observation mismatch in lieu of likelihood. For multivariate data, there are significant difficulties in how to define this measure. The algorithm described here is very different from ABC.

In fact, this algorithm is not intrinsically tied to the anchor parameterization or anchored inversion. It is a general algorithm for approximating the posterior distribution where the likelihood is unknown but the model (which corresponds to the likelihood) can be simulated.

5 Automatic and adaptive choice of anchors

The definition of anchors in (3) is represented by the matrix \mathbf{H} . There are two aspects in the choice of anchors, namely the number of anchors (i.e. the number of rows in \mathbf{H}) and the specific definition of each anchor (i.e. the content of each row of \mathbf{H}). In this section we develop a procedure that adjusts both aspects in an automatic, adaptive manner, taking advantage of the iterative nature of the algorithm.

We partition the model domain (or numerical grid) into a number of (equal- or unequal-sized) subsets and define the mean value of Y in each subset as an anchor. These anchors form a *partitioning anchorset*, or “anchorset” for short. Each anchorset is defined by a unique matrix \mathbf{H} as introduced before. This definition is instrumental in the adaptive procedure, to be presented shortly, which examines a number of anchorsets and chooses the one that has the best predicted performance in the upcoming iteration.

5.1 A measure of model performance

The choice of anchorset is a model selection problem, because any anchorset represents a particular parameterization, or “model”, for the field $Y(x)$. To facilitate comparison of models, we measure the performance of a model by a kind of integrated predictive likelihood defined as

$$L_I = \int_{\Theta} p(z | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) f^{(k)}(\theta) d\theta,$$

where $f^{(k)}(\theta)$ is the approximate posterior density of θ obtained in the k th iteration. This measure is analogous to Bayes factors [Kass and Raftery, 1995; George, 2006] or, in particular, “posterior” Bayes factors [Aitkin, 1991]. However, the current setting differs from common (Bayesian) model selection in several ways. For example, the likelihood $p(z | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is unknown, and is being estimated; the ultimate subject is not really the model parameter θ , but rather the field Y ; the algorithm is iterative, which makes use of the data z in each iteration and corrects for the repeated use of data by importance weighting.

As an approximation, we assume the predictive distribution, $\int p(z | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) f^{(k)}(\theta) d\theta$, is normal. The predictions z_1, \dots, z_n generated in step 2 of the algorithm is a random sample of this distribution. Based on the sample mean \bar{z} and sample covariance matrix \mathbf{S}_z of z_1, \dots, z_n , an approximation of $\log L_I$ could be computed by

$$L = \log \phi(z | \bar{z}, \mathbf{S}_z),$$

where z is the observation.

The measure L is subject to a fair amount of uncertainty arising from several sources. First, it is computed based on *random samples* $\theta_1, \dots, \theta_n$ and z_1, \dots, z_n . Second, each z is a random draw conditional on its corresponding θ (via a random draw of \mathbf{y}). Third, the normal assumption for Z may not be fully justifiable. Fourth, the sample size (n) is rather moderate in view of the dimensions of the random variables (n_θ and n_z). In experiments we used n in the thousands and $n_\theta + n_z$ in the tens or hundreds. In fact, when $n_\theta + n_z$ is relatively large (say about 100), it is not rare that the empirical covariance \mathbf{S}_z is not invertible. In view of

these subtleties that harm the robustness of L , we further simplify and take

$$L^* = \sum_{j=1}^{n_z} \log \phi(z[j] \mid \overline{z[j]}, s_{z[j]}^2), \quad (14)$$

where $[j]$ indicates the j th dimension of Z . This approximation ignores correlations between the components of Z .

5.2 Predicting the performance of alternative anchorsets

Denote the anchorset at the beginning of the k th iteration by \mathbf{H} . Towards the end of this iteration (somewhere in step 5), we consider whether to switch to an alternative anchorset, say \mathbf{H}_* , in the next iteration. The decision to switch or not will be made after comparing the *predicted performances* of the model in the next iteration using anchors \mathbf{H} and \mathbf{H}_* , respectively. Let the two performances be denoted by $L_{k+1}^*(\mathbf{H})$ and $L_{k+1}^*(\mathbf{H}_*)$, respectively.

The model (5), the prior (7), and the relation (6) suggest that θ_* conditional on θ has a normal distribution, which we denote by $\pi(\theta_* \mid \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ to stress the fact that it is derived from the “prior” relations. Analogously, we write $\pi(\theta \mid \theta_*, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ and $\pi(\theta, \theta_* \mid \boldsymbol{\mu}, \boldsymbol{\Sigma})$.

The distribution $f^{(k-1)}(\theta)$ induces a distribution for θ_* , which we denote by $f_*^{(k-1)}(\theta_*)$. In fact, the sampling in steps 1–2 of the k th iteration also generates a sample of the alternative anchors, $\{\theta_{*i} = \mathbf{H}_* \mathbf{y}_i\}_{i=1}^n$, and the sample is a random sample of $f_*^{(k-1)}(\theta_*)$ by the definition of $f_*^{(k-1)}(\theta_*)$. Alternatively, we can consider the sample $\{\theta_{*i}\}$ to be obtained via $f_*^{(k-1)}(\theta_*) = \int_{\Theta} \pi(\theta_* \mid \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) f^{(k-1)}(\theta) d\theta$.

Suppose we continue to use \mathbf{H} in the $(k+1)$ th iteration. In step 5 of the k th iteration, $f^{(k-1)}(\theta)$ is updated to $f^{(k)}(\theta)$. The distribution $f^{(k)}(\theta)$ will be used in steps 1–2 of the $(k+1)$ th iteration to generate a random sample of Z , which is then used to calculate the performance indicator $L_{k+1}^*(\mathbf{H})$. Due to the high computational cost of the forward model, however, we do not want to actually generate this sample of Z before committing to using the anchors \mathbf{H} in the $(k+1)$ th iteration. The idea then is to use the sample z_1, \dots, z_n created in step 2 of the k th iteration, which is based on $f^{(k-1)}(\theta)$, as an importance sample from the distribution of Z that would be generated in step 2 of the $(k+1)$ th iteration, which would be based on $f^{(k)}(\theta)$. The importance weights are proportional to $f^{(k)}(\theta_i) / f^{(k-1)}(\theta_i)$, $i = 1, \dots, n$. This weighted sample of Z then provides weighted sample means $\overline{z[j]}$ and weighted sample variances $s_{z[j]}^2$ to be used in (14) for predicting $L_{k+1}^*(\mathbf{H})$.

Now suppose we switch to \mathbf{H}_* in the $(k+1)$ th iteration. First, at the end of step 5 of the k th iteration, we need to obtain $f_*^{(k)}(\theta_*)$ instead of $f^{(k)}(\theta)$. Second, the distribution $f_*^{(k)}(\theta_*)$ would give rise to a sample of Z in step 2 of the $(k+1)$ th iteration, but we again would like to use the sample z_1, \dots, z_n created in step 2 of the k th iteration as a *weighted* replacement. We address these two steps now.

- (1) We would like to view the sample $\{\theta_{*i}\}$ as an importance sample from $\pi(\theta_*)$, then we would be able to conduct steps 4–5 for θ_* (instead of θ) and obtain $f_*^{(k)}(\theta_*)$. The question is: What are the importance weights?

If the anchorsets \mathbf{H} and \mathbf{H}_* have no common anchors, then

$$\pi(\theta_*) = \int_{\Theta} \pi(\theta_* \mid \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \pi(\theta) d\theta = \int_{\Theta} \pi(\theta_* \mid \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \frac{\pi(\theta)}{f^{(k-1)}(\theta)} f^{(k-1)}(\theta) d\theta. \quad (15)$$

This suggests that the importance weights are proportional to $\frac{\pi(\theta)}{f^{(k-1)}(\theta)}$.

If \mathbf{H} and \mathbf{H}_* share some common anchors, then write $\theta = (\theta_\square, \theta_\diamond)$ and $\theta_* = (\theta_\square, \theta_\star)$. Let $f^{(k-1)}(\theta_\square)$ and $f^{(k-1)}(\theta_\diamond | \theta_\square)$ denote the marginal and conditional distributions induced by $f^{(k-1)}(\theta)$. We have

$$\begin{aligned} \pi(\theta_*) &= \pi(\theta_\square) \pi(\theta_\star | \theta_\square, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \\ &= f^{(k-1)}(\theta_\square) \frac{\pi(\theta_\square)}{f^{(k-1)}(\theta_\square)} \int_{\Theta_\diamond} \pi(\theta_\star | \theta_\diamond, \theta_\square, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \frac{\pi(\theta_\diamond | \theta_\square, \boldsymbol{\mu}, \boldsymbol{\Sigma})}{f^{(k-1)}(\theta_\diamond | \theta_\square)} f^{(k-1)}(\theta_\diamond | \theta_\square) d\theta_\diamond. \end{aligned} \quad (16)$$

Noticing that the sample $\{\theta_{*i}\}$ is obtained via $f^{(k-1)}(\theta_\square) f^{(k-1)}(\theta_\diamond | \theta_\square) \pi(\theta_\star | \theta_\diamond, \theta_\square, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, the above suggests the importance weights are proportional to $\frac{\pi(\theta_\square)}{f^{(k-1)}(\theta_\square)} \frac{\pi(\theta_\diamond | \theta_\square, \boldsymbol{\mu}, \boldsymbol{\Sigma})}{f^{(k-1)}(\theta_\diamond | \theta_\square)} = \frac{\pi(\theta)}{f^{(k-1)}(\theta)}$.

Relations (15)–(16) show that the weights are proportional to $\pi(\theta_i)/f^{(k-1)}(\theta_i)$ when $\{\theta_{*i}\}$ are taken as an importance sample from $\pi(\theta_*)$. These are the same weights (i.e. the w_i s) had we continued to use θ in steps 4–5. In other words, the weights do not depend on the specific choice of \mathbf{H}_* . Using these weights and the sample $\{\theta_{*i}, z_i\}$, steps 4–5 of the k th iteration produce $f_*^{(k)}(\theta_*)$.

- (2) After deriving $f_*^{(k)}(\theta_*)$, we would view $\{z_i\}$, obtained in step 2 of the k th iteration, as an importance sample of the Z that would be generated starting with $f_*^{(k)}(\theta_*)$ in steps 1–2 of the $(k+1)$ th iteration. The importance weights are equal to those of $\{\theta_{*i}\}$ obtained in the k th iteration as an importance sample from $f_*^{(k)}(\theta_*)$.

If the anchorsets \mathbf{H} and \mathbf{H}_* have no common anchors, then

$$\begin{aligned} f_*^{(k)}(\theta_*) &= \frac{f_*^{(k)}(\theta_*)}{\pi(\theta_*)} \int_{\Theta} \pi(\theta_* | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \pi(\theta) d\theta \\ &= \frac{f_*^{(k)}(\theta_*)}{\pi(\theta_*)} \int_{\Theta} \pi(\theta_* | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \frac{\pi(\theta)}{f^{(k-1)}(\theta)} f^{(k-1)}(\theta) d\theta. \end{aligned} \quad (17)$$

This suggests the importance weights are proportional to

$$\frac{f_*^{(k)}(\theta_*)/\pi(\theta_*)}{f^{(k-1)}(\theta)/\pi(\theta)}. \quad (18)$$

If \mathbf{H} and \mathbf{H}_* share some common anchors, then, using the notation introduced above, we have

$$\begin{aligned} f_*^{(k)}(\theta_*) &= f_*^{(k)}(\theta_\square) f_*^{(k)}(\theta_\star | \theta_\square) \\ &= f^{(k-1)}(\theta_\square) \frac{f_*^{(k)}(\theta_\square)}{f^{(k-1)}(\theta_\square)} \frac{f_*^{(k)}(\theta_\star | \theta_\square)}{\pi(\theta_\star | \theta_\square)} \int_{\Theta_\diamond} \pi(\theta_\star | \theta_\diamond, \theta_\square, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \pi(\theta_\diamond | \theta_\square) d\theta_\diamond \\ &= f^{(k-1)}(\theta_\square) \frac{f_*^{(k)}(\theta_\square)}{f^{(k-1)}(\theta_\square)} \frac{f_*^{(k)}(\theta_\star | \theta_\square)}{\pi(\theta_\star | \theta_\square)} \\ &\quad \int_{\Theta_\diamond} \pi(\theta_\star | \theta_\diamond, \theta_\square, \boldsymbol{\mu}, \boldsymbol{\Sigma}) \frac{\pi(\theta_\diamond | \theta_\square)}{f^{(k-1)}(\theta_\diamond | \theta_\square)} f^{(k-1)}(\theta_\diamond | \theta_\square) d\theta_\diamond, \end{aligned} \quad (19)$$

suggesting the importance weights are proportional to

$$\frac{f_*^{(k)}(\theta_\square)}{f^{(k-1)}(\theta_\square)} \frac{f_*^{(k)}(\theta_\star | \theta_\square)}{\pi(\theta_\star | \theta_\square)} \frac{\pi(\theta_\diamond | \theta_\square)}{f^{(k-1)}(\theta_\diamond | \theta_\square)} = \frac{f_*^{(k)}(\theta_*)/\pi(\theta_\star | \theta_\square)}{f^{(k-1)}(\theta)/\pi(\theta_\diamond | \theta_\square)} = \frac{f_*^{(k)}(\theta_*)/\pi(\theta_*)}{f^{(k-1)}(\theta)/\pi(\theta)}. \quad (20)$$

The weights turn $\{z_i\}$ into a weighted sample in computing $\overline{z[j]}$ and $s_{z[j]}^2$ for predicting $L^*(\mathbf{H}_*)$ by (14).

Next, if it turns out that $L^*(\mathbf{H}_*) > L^*(\mathbf{H})$, we take $f_*^{(k)}(\theta_*)$ as the output of the k th iteration, with the anchorset \mathbf{H}_* . Otherwise, the output is $f^{(k)}(\theta)$ with the anchorset \mathbf{H} .

A remaining question is, how do we identify, in a way that is flexible and efficient, candidate alternative anchorsets (i.e. \mathbf{H}_* 's) to examine?

5.3 A strategy for defining alternative anchorsets

Suppose the current anchorset, represented by \mathbf{H} , contains n_θ anchors. Recall the definition of anchorset: the model domain is partitioned into n_θ sub-regions; the mean value of Y in each sub-region is designated an anchor. Let us call each sub-region the “support” of the corresponding anchor.

We consider n_θ alternative anchorsets, denoted by $\mathbf{H}_1, \dots, \mathbf{H}_{n_\theta}$, each containing $n_\theta + 1$ anchors. The anchorset \mathbf{H}_j , $j = 1, \dots, n_\theta$, is identical to the original anchorset \mathbf{H} except that the support of the j th anchor in \mathbf{H} is split into two sub-regions, defining two new anchors.

The $2n_\theta$ newly-created “small” anchors resulting from splitting each anchor in \mathbf{H} form an “umbrella” anchorset, denoted by \mathbf{H}_* , which defines anchor vector $\theta_* = \mathbf{H}_* \mathbf{Y}$. Similarly defined are anchor vectors $\theta = \mathbf{H} \mathbf{Y}$, $\theta_1 = \mathbf{H}_1 \mathbf{Y}, \dots, \theta_{n_\theta} = \mathbf{H}_{n_\theta} \mathbf{Y}$. It can be seen that each of $\theta, \theta_1, \dots, \theta_{n_\theta}$ is a linear function of θ_* . In steps 3–5 of the algorithm, we estimate the density of θ_* . (To estimate $f^{(k)}(\theta_*)$, extract the sample $\{\mathbf{H}_* \mathbf{y}_i\}$, then use the weights $\{w_i\}$ in steps 3–5.) Because the estimate is a normal mixture, the densities of $\theta, \theta_1, \dots, \theta_{n_\theta}$ follow immediately. Based on these densities, we predict the L^* for each of these $n_\theta + 1$ anchorsets and choose the anchorset that achieves the largest L^* . Steps 3–5 of the algorithm are then repeated to estimate $f^{(k)}$ for the chosen anchorset.

By this strategy, steps 3–5 of the algorithm are performed twice in order to compare and choose from $n_\theta + 1$ anchorsets, including the original and the alternatives. As iterations proceed, more alternative anchorsets are considered in each iteration as the size of (i.e. number of anchors contained in) the currently used anchorset grows. The computational cost, however, barely increases. This efficient strategy takes advantage of the fact that the weights $\{w_i\}$ remain the same for all anchorsets.

An obvious extension to this strategy allows an alternative anchorset to split more than one anchor (say up to two) in the original anchorset. This extension examines more alternative anchorsets without substantially increasing the computational cost, because the umbrella anchorset \mathbf{H}_* remains unchanged.

As this adaptive procedure suggests, the inversion exercise begins with a low number of anchors and gradually increases the number of anchors in iterations. In particular, we may begin with an anchorset that bisects each dimension of the space; that is, use 2, 4, and 8

anchors in 1D, 2D, and 3D spaces, respectively. This completely relieves the user from the burden of specifying an initial anchorset. More anchors are introduced in the iterations as more computation has been invested in the task. New anchors tend to be introduced in regions of the model domain where a more detailed representation of the Y field is predicted to bring about most improvement to the model's capability in reproducing the observation z . The partition of the model domain by the anchorset is not uniform—it is tailored to the unknown field Y , the forward process \mathcal{M} , and existing anchors in an automatic, adaptive, and evolving fashion.

5.4 Modifications to the algorithm in section 4.1

In steps 3–5, the original θ is replaced by the umbrella θ_* (obtained from the sample $\{\mathbf{y}_i\}$ via $\mathbf{H}_*\mathbf{y}_i$), which is usually twice as long as θ . At the end of step 5, $p(\theta_* | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is obtained.

Step 5 now does not conclude an iteration. Display (13) is followed by the following new step.

6. *Selecting an anchorset and deriving its distribution.*

From the estimated density $p(\theta_* | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, given by (13), derive the density of each alternative anchorset as well as the original anchorset, then predict L^* for each of these anchorsets.

Pick the anchorset that achieves the largest L^* . Use this anchorset to repeat steps 3–5 and derive its density $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$. This density becomes $f^{(k)}(\theta)$, where θ corresponds to the picked anchorset.

This concludes an iteration of the algorithm.

6 Extensions

6.1 Geostatistical parameterization of the field

The mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$ of the field may be parameterized in conventional geostatistical forms. Because of the existence of anchors, sophisticated trend models, such as polynomials in spatial coordinates, are usually unnecessary. One may model the mean $\boldsymbol{\mu}$ by a global constant:

$$\boldsymbol{\mu} = \beta.$$

The stationary covariance may be parameterized as

$$\text{cov}(Y(x_1), Y(x_2)) = \eta^2 \rho(x_1 - x_2; \varphi),$$

where η^2 is variance and φ contains parameters pertaining to range, smoothness, nugget, geometric anisotropy, and so on. We leave specifics of this parameterization open. See section 7.1 for an example of a particular form of this parameterization.

A requirement imposed by the proposed inversion method is that all unknown parameters are defined on supports that can be mapped onto $(-\infty, \infty)$. This usually means that the support of each parameter should be in one of the following forms: $(-\infty, \infty)$, $(-\infty, c)$, (c, ∞) , and (c_1, c_2) . Once this requirement is satisfied, all parameters are transformed to

be defined on $(-\infty, \infty)$, hence the posterior distribution can be approximated by a normal mixture. Note the support of each anchor is already $(-\infty, \infty)$ because the support of Y is $(-\infty, \infty)$.

Let us denote the geostatistical parameters β , η^2 , and φ , transformed onto $(-\infty, \infty)$, by θ_1 and re-label the anchor parameters θ_2 . The complete parameter vector is now $\theta = (\theta_1, \theta_2)$.

The parameterization of Y described in section 3, or (6) in particular, now is written as $p(\mathbf{y}|\theta)$, i.e. $p(\mathbf{y}|\theta_1, \theta_2)$, in which θ_1 determines $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, and θ_2 plays the role of the θ in (6). The expanded parameter vector retains the simplicity of a geostatistical description of the random field. In the meantime it enables, via (6), sophisticated mean and covariance structures for Y beyond the expressing capability of the geostatistical parameters alone. To some extent, one may say that the geostatistical parameters capture “global” features, whereas the anchor parameters capture “local” features.

A prior for θ can be specified by

$$\pi(\theta) = \pi(\theta_1) \pi(\theta_2 | \theta_1), \quad (21)$$

where $\pi(\theta_1)$ is a prior for the geostatistical parameters, and $\pi(\theta_2 | \theta_1) = \phi(\theta_2 | \mathbf{H}\boldsymbol{\mu}, \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T)$ in which $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are determined by θ_1 . The posterior is now written as $p(\theta | z)$ instead of $p(\theta | z, \boldsymbol{\mu}, \boldsymbol{\Sigma})$.

In the algorithm in section 4, all the θ ’s should be understood as the expanded parameter vector, $\theta = (\theta_1, \theta_2)$. The overall idea of the algorithm can be summarized as “simulate $p(\theta_1, \theta_2, Z)$, then condition on the observed z .”

In step 1 of the algorithm, the prior $\pi(\theta_i)$ is now given by (21). In step 2, $p(\mathbf{y} | \theta_i, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is now $p(\mathbf{y} | \theta_i)$, in which θ_i contains both anchor parameters and geostatistical parameters, the latter providing $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$.

Similarly in section 5, the θ and θ_* should be understood as the extended parameter vectors, and their priors are given by (21). In the meantime $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ should be dropped because they are now determined by θ_1 . Note that the θ_1 part is not affected by the search for alternative θ_2 , hence θ_1 plays the role of the θ_\square in (16), (19), and (20).

6.2 Using linear data

Besides the nonlinear data z , one may have some direct measurements of the spatial attribute Y itself or, more generally, measurements of some linear function of Y . In the groundwater example in section 1, linear data may include direct measurements of local-scale hydraulic conductivity and covariates such as grain-size distribution and core-support geophysical properties, which provide estimates of local hydraulic conductivity via empirical relations. In the geophysical example in section 1, linear data may be mechanic properties of the elastic medium at the ground surface. In the atmospheric example in section 1, linear data may be direct monitoring of CO_2 sources on the ground surface or covariates that provide estimates of CO_2 sources.

Linear data enter the proposed inversion procedure at two places: (1) the prior of anchors, $\pi(\theta)$, not only relies on the mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$, but is also conditioned on the linear data; (2) while simulating field realizations conditional on values of the anchor parameters, the simulation is in addition conditioned on the linear data. The modified prior can be

alternatively viewed as the prior $\pi(\theta)$, ignoring the linear data, multiplied by the likelihood of θ with respect to the linear data.

Let the linear data be denoted by $\ell = \mathbf{L}\mathbf{y}$. (Naturally, \mathbf{L} shares no common row with \mathbf{H} , the definition of anchors.) If the geostatistical extension described in section 6.1 is not used, the prior of θ given by (7) is replaced by

$$\pi(\theta) \propto p(\theta | \boldsymbol{\mu}, \boldsymbol{\Sigma}) p(\ell | \theta, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = p(\theta, \ell | \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \phi\left(\begin{bmatrix} \theta \\ \ell \end{bmatrix} \mid \begin{bmatrix} \mathbf{H} \\ \mathbf{L} \end{bmatrix} \boldsymbol{\mu}, \begin{bmatrix} \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^T & \mathbf{H}\boldsymbol{\Sigma}\mathbf{L}^T \\ \mathbf{L}\boldsymbol{\Sigma}\mathbf{H}^T & \mathbf{L}\boldsymbol{\Sigma}\mathbf{L}^T \end{bmatrix}\right). \quad (22)$$

The conditional distribution of the field \mathbf{Y} given the anchor values is now $p(\mathbf{y} | \theta, \ell, \boldsymbol{\mu}, \boldsymbol{\Sigma})$. This can be handled by combining θ and ℓ into a single set of linear condition and using a formula analogous to (6).

In this situation, the target of the exercise is the posterior $p(\theta | z, \ell, \boldsymbol{\mu}, \boldsymbol{\Sigma})$. The overall idea of the algorithm can be summarized as “simulate $p(\theta, Z | \ell, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, then condition on the observed z .”

If the geostatistical extension is in place, let us denote $\theta = (\theta_1, \theta_2)$ as in section 6.1. The prior in (21) is replaced by

$$\pi(\theta) \propto \pi(\theta_1) p(\theta_2 | \theta_1) p(\ell | \theta_1, \theta_2) = \pi(\theta_1) p(\theta_2, \ell | \theta_1). \quad (23)$$

The conditional $p(\theta_2, \ell | \theta_1)$ is normal, analogous to (22) except that $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are determined by θ_1 . The conditional distribution of \mathbf{Y} given the model parameter θ is conditioned on ℓ in addition to θ , that is, $p(\mathbf{y} | \theta, \ell) = p(\mathbf{y} | \theta_2, \theta_1, \ell)$, where θ_2 and ℓ are linear conditions, and θ_1 provides $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. This is again similar to (6).

In this situation, the target of the exercise is the posterior $p(\theta_1, \theta_2 | z, \ell)$. The overall idea of the algorithm can be summarized as “simulate $p(\theta_1, \theta_2, Z | \ell)$, then condition on the observed z .”

7 Examples

We will present three examples of scientific applications using synthetic data. The first two examples are in one-dimensional space, whereas the third is in two-dimensional space. In all three examples, the unknown physical property of interest is positive by definition, hence we take Y to be the logarithm of the physical property, and treat Y as a Gaussian process. The examples demonstrate a number of aspects of the anchored inversion methodology. In example 1, the forward data are measurements distributed in space. In example 2, the forward data are time-series measurements at a boundary point of the model domain. In example 3, the forward data are neither time series nor attached to specific spatial locations. In fact, all components of the forward-data in the examples are functions of the entire field, even if their measurement is associated with specific locations or times. Example 1 also uses a linear datum (i.e., a direct measurement of Y at one location). In example 2, the forward function is a composition of two separate field processes, giving two datasets (to be combined to form the forward data vector) that are of incomparable physical natures. In example 3, the inversion algorithm is executed in a numerical grid that is coarser than the synthetic true field, introducing systematic model error. In all examples, there exists substantial correlation between components of the forward data (i.e. variable Z).

In all the examples, the data were assumed to be error-free. (In example 3, we knew there was model error but chose to ignore it.) Because the true field is known in these synthetic examples, field simulations compared with the true field demonstrate the performance of the inversion. Another assessment comes from comparing predictions of the forward data in simulated fields with the observed forward data.

In all examples, the algorithm was run for 20 iterations with sample sizes (the ns) 2400, 1950, 1612, ..., 610, 608, totaling 19176 samples (which is also the total number of forward model runs). The initial anchors are designated by evenly bisecting each spatial dimension (hence the algorithm starts with 2 and 4 anchors in 1-D and 2-D examples, respectively). As iteration proceeds, anchor designations evolve automatically as described in section 5. In each iteration we considered splitting one of the existing anchors, hence the number of anchors increased by 0 or 1 in any single iteration.

Before delving into individual examples, in sections 7.1–7.2 we address some general preparations that are used in all examples. In these two sections, both Y and Z are generic symbols for the spatial variable and forward data, respectively, as the actual meanings of both depend on the specific example.

7.1 Geostatistical parameterization, prior specification, and initial approximation

For all the examples, we used a conventional geostatistical formulation for the field mean μ and covariance Σ . Following the notation in section 6.1, the geostatistical parameters, anchor parameters, and the complete parameter vector are denoted by θ_1 , θ_2 , and $\theta = (\theta_1, \theta_2)$, respectively. The same procedures were followed in all examples to specify the prior for θ_1 and construct the initial approximation $f^{(0)}$. Recall that the methodology is open with respect to these elements. However, the specific choices made for these examples are sensible starting points in general applications.

The parameterization of the field Y (before conditioning on anchors) includes a global mean β and a covariance function

$$\begin{aligned} \text{cov}(Y(x_1), Y(x_2)) &= (1 - \tau)\eta^2\rho(|x_1 - x_2|) + \tau\eta^2I(x_1 = x_2) \\ &= (1 - \tau)\eta^2\left(1 + \frac{|x_1 - x_2|}{\lambda}\right) \exp\left(-\frac{|x_1 - x_2|}{\lambda}\right) + \tau\eta^2I(x_1 = x_2), \end{aligned} \quad (24)$$

in which η^2 is variance, $\tau \in [0, 1)$ is “nugget”, $I(\cdot)$ is the indicator function assuming the value 1 if its arguments is true and 0 otherwise, λ is “range” or “scale”, and ρ is an isotropic correlation function. The particular form of ρ adopted above is a special case of the Matérn correlation function with fixed smoothness $\kappa = 1.5$. This correlation function has been used by [Zhang et al. \[2007\]](#) in modeling hydrodynamics. In-depth discussion on the Matérn family of correlation functions can be found in [Stein \[1999\]](#). In sum, the geostatistical formulation uses a parameter vector $(\beta, \lambda, \eta^2, \tau) \in (-\infty, \infty) \times (0, \infty) \times (0, \infty) \times [0, 1)$, in the natural unit of each component. (Geometric anisotropy and the option of taking κ as an unknown parameter are also implemented.) The following prior is adopted for these parameters (see [Zhang \[2012\]](#) for details):

$$\pi(\beta, \lambda, \eta^2, \tau) = \text{gamma}(\lambda; \cdot) (\eta^2)^{-1} \text{beta}(\tau; \cdot).$$

To use the proposed algorithm, each parameter component is transformed onto $(-\infty, \infty)$, leading to the geostatistical parameter vector

$$\theta_1 = \left(\beta, \log \lambda, \log \eta^2, \log \left(\frac{\tau}{1 - \tau} \right) \right).$$

The prior of θ_1 is determined by the prior above (which is in terms of the parameters in their “natural” units) and the transformations. This prior is used in (21) or (23) (depending on the presence or absence of linear data) to specify a prior for the full model parameter θ . Note that a prior for the anchor parameters, θ_2 , is determined by (21) or (23); no user intervention is required.

The initial approximation $f^{(0)}(\theta)$ was taken to be a multivariate normal distribution that is fairly diffuse.

7.2 Diagnostics

Roughly speaking, the integrated likelihood L^* defined by (14) measures how well the model reproduces, or predicts, the observations. Here, “reproduction” refers to the output of the forward model evaluated at Y fields simulated according to the posterior distribution of θ . In real-world applications, a comparison of such model predictions with the actual observations provides one of the more concrete assessments of the model. An increasing L^* in iterations suggests the model is improving. Hence L^* can be monitored as a diagnostic.

Another summary of this prediction-observation comparison is as follows. Consider the sample $\{z_1, \dots, z_n\}$ created in step 2 of the very first iteration (that is, based on $f^{(0)}(\theta)$). For dimension j of Z , where $j = 1, \dots, n_z$, compute the median absolute difference between the sample and the observation, and denote it by $\text{mad}_{[j]}^{(0)}$. Later in the i th iteration, compute $\text{mad}_{[j]}^{(i)}$ and obtain the ratio $r_{\text{mad}}^{(i)[j]} = \text{mad}_{[j]}^{(i)} / \text{mad}_{[j]}^{(0)}$. This “mad ratio” for each dimension of Z is expected to decrease in iterations until it reaches a stable level. In each iteration, summaries such as the median and the maximum of the n_z mad ratios are reported as diagnostics.

As a diagnostic, the “mad ratio” is bounded below by 0 and rarely exceeds 1. This makes it easier to interpret than L^* .

7.3 Example 1: groundwater flow

Consider the groundwater flow example mentioned in section 1 in one-dimensional space. Denote hydraulic conductivity by K (ms^{-1}) and hydraulic head by h (m). We took $Y = \log K$ (because K is positive) and modeled Y as a one-dimensional Gaussian process. The synthetic log-conductivity field is shown in figure 1. The field is composed of 100 grid points; this constitutes the model domain.

The one-dimensional, steady-state groundwater flow in saturated zone is described by the following differential equation [Schwartz and Zhang, 2003, chap. 5]:

$$\frac{d}{dx} \left(K \frac{dh}{dx} \right) = 0, \tag{25}$$

where x is spatial coordinate. Here we have assumed there is no water source or sink in the model domain (but there may be water injection or extraction at the boundaries in order

to maintain certain boundary conditions). Both K and h vary with x . We view K as the “controlling” physical attribute and h as the “outcome”. Given the 1-D conductivity field K , we can solve this equation to find the head h everywhere in the model domain. The scientific question is to infer K given measured h at a number of locations.

Suppose h at 30 locations in the field were available as forward data. The forward model \mathcal{M} involved transforming Y to $K = e^Y$, solving (25) for h under specified boundary conditions (that is, $h = 1$ at the left boundary and $h = 0$ at the right boundary), and extracting the 30 values of h at the measurement locations. The synthetic forward data (figure 2) are the output of the forward model given the synthetic field as input.

Noticing in (25) that a scaling of K (or a shifting of Y) does not change the resultant h , we need at least one direct measurement of Y . The synthetic Y value at a randomly picked location, marked in figure 1, was made available to the inversion as linear data.

Some summaries and diagnostics of the result are listed in table 1. In the table one sees decisive increase in L^* and decrease in mad ratios, as desired. Overall, the number of anchors started at 2 and increased to 16. Because the geostatistical parameter had 4 components, the dimension of the model parameter θ started at 6 and increased to 20. The adaptive nature of anchor selection in the algorithm is vividly depicted in figure 3, which shows the evolution of the anchorsets in the iterations.

Figure 4 shows the inferred marginal distribution of each anchor, as represented by a boxplot of the sample obtained in step 1 of the algorithm. Since the anchorsets evolved in iterations, we do not see the “convergence” of the distribution of a fixed anchorset. Because the true field is known, each anchor has a “target value” defined by the true Y values. These target values are indicated in figure 4. It is seen that the estimated distributions homed in on the target values convincingly after a few iterations. This strength of convergence for a reasonably large number of anchors suggests that the parameterization provides a good presentation of the field.

The estimated marginal distributions of the geostatistical parameters are shown in figure 5. It should be stressed that the meaning of the geostatistical parameters changes during the iterations whenever the anchorset changes. In addition, interactions between the geostatistical parameters cause difficulty in their inference (see Zhang [2012]). Compared to the anchor parameters, the role of geostatistical parameters is more of an intermediate modeling device.

Recall that the goal of anchored inversion is to characterize the field Y . We created 1000 field realizations based on the anchorset and approximate posterior distribution in the final iteration (i.e. $f^{(20)}(\theta)$ in this particular example). The 5th, 25th, 50th, 75th, and 95th percentiles of the realizations for each grid point are plotted in figure 6, overlaid with the synthetic field for comparison. The comparison confirms the inversion worked well.

In real-world applications, we do not know the true field, therefore can neither compare the field simulations with the truth nor compare the anchors with their target values. Perhaps the most concrete assessment is provided by the capabilities of simulated fields to “reproduce” the forward data. In each iteration of the algorithm, the sample $\{z_i\}_1^n$ obtained in step 2 represents “predictions” of the observation based on the posterior distribution estimated in the previous iteration. These predictions are compared with the actual observations in figure 7. In each panel (for one iteration), a vertical box-plot for each component of the Z predictions is drawn on the horizontal location where the observed value is. Therefore, if

Table 1: Summaries and diagnostics for Example 1.

Iteration	Sample size	Anchors	L^*	Median $r_{\text{mad}}^{(i)[j]}$	Max $r_{\text{mad}}^{(i)[j]}$
1	2400	2	-48.6	1.0000	1.000
2	1950	3	-7.4	0.1829	0.346
3	1612	4	31.7	0.0354	0.188
4	1359	5	35.2	0.0402	0.146
5	1170	6	43.3	0.0290	0.138
6	1027	7	49.1	0.0270	0.086
7	920	8	72.9	0.0110	0.089
8	840	9	85.1	0.0081	0.063
9	780	10	84.0	0.0095	0.034
10	735	11	83.6	0.0091	0.034
11	701	11	70.0	0.0086	0.040
12	676	11	85.8	0.0096	0.025
13	657	11	90.4	0.0063	0.026
14	643	12	96.0	0.0064	0.022
15	632	13	99.2	0.0051	0.023
16	624	14	98.8	0.0052	0.021
17	618	15	103.3	0.0040	0.036
18	614	15	106.0	0.0037	0.037
19	610	15	101.5	0.0038	0.038
20	608	16	97.8	0.0037	0.038

the predictions reproduce the observations, the box-plots should fall centered on the $y = x$ line. The figure demonstrates that the reproduction starts severely biased and outspread in the initial iteration and, after steady adjustment of the mean and reduction of the variance in subsequent iterations, achieves precise match with the actual observation in the final iterations.

7.4 Example 2: rainfall and surface runoff

Runoff over the land surface generated during and after a rainfall event is affected by the hydraulic roughness of the land surface [Vieux, 2004, chap 6]. By affecting runoff, the roughness also exerts important influence on infiltration, erosion, and vegetative growth. However, detailed measurement of the roughness coefficient in a watershed is impractical. In many hydrological and agricultural models, surface roughness is treated as a constant or as a categorical variable taking a few values based on the type of the land cover. Huang and Lee [2009] investigate the effect of spatially heterogeneous roughness on the runoff process. They consider a one-dimensional overland plane model described by the following equation

system:

$$\frac{\partial h}{\partial t} + \frac{\partial q}{\partial x} = b, \quad s_f = s_0 - \frac{\partial h}{\partial x}, \quad q = hu, \quad u = r^{-1}h^{2/3}s_f^{1/2} \quad (26)$$

where x is the spatial coordinate (m), t is the time (s), $h(x, t)$ is the flow depth (m), $q(x, t)$ is the flow discharge per unit width (m^2s^{-1}), $b(x, t)$ is the effective rainfall (ms^{-1}), $s_f(x, t)$ is the friction slope (m/m), $s_0(x)$ is the bed slope (m/m), $u(x, t)$ is the flow velocity (ms^{-1}), and $r(x)$ is the Manning roughness coefficient ($\text{sm}^{-1/3}$). In this system, the rainfall input b and the bed slope s_0 are considered known. In addition, the initial condition $h(x, 0) = q(x, 0) = 0$ and boundary condition $q(0, t) = 0$ are assumed. These conditions state that the surface is dry at the beginning of the rainfall event, and there is no inflow at the upstream boundary.

Whereas Huang and Lee [2009] investigate the resultant h and q under various configurations of synthetic roughness $r(x)$, we considered the inverse problem: infer the spatial field of $r(x)$ given observations of h and q . The synthetic log-roughness field we used is shown in figure 8. The field consists of 150 grid points.

To make it more interesting, we constructed two synthetic rainfall events and corresponding observation scenarios. In the first event, rainfall lasted for 60 minutes with a uniform intensity of 5 mm/hr. The discharge at the downstream boundary was measured every 10 minutes for 3 hours (that is, during the rain and in the subsequent 120 minutes). In the second event, rainfall lasted for 30 minutes with a uniform intensity of 20 mm/hr. The flowdepth at the downstream boundary was measured every 5 minutes for 4 hours (that is, during the rain and in the subsequent 210 minutes). These two time-series datasets were combined to form a forward data vector of length 66. The discharge and flowdepth processes at the downstream boundary in the two rain events, computed based on the synthetic roughness field, are depicted in figure 9. We took $Y = \log r$ and modeled Y as a Gaussian process. The forward function \mathcal{M} involved obtaining $r = e^Y$, solving the equation system (26) for h and q based on each of the two rainfall scenarios, extracting q (or h) at the specified space and time coordinates, and taking the combined vector $(\log q, \log h)$ as the forward data z . Note here the forward data contained some discharge measurements and some flowdepth measurements. These two physical quantities are of incomparable natures.

In the iterative algorithm, principal component analysis reduced the dimension of the forward data from 66 down to 9–19. Such significant dimension reduction was expected given the high level of auto-correlation in the time series data.

Performance of the inversion is demonstrated by 1000 field realizations created based on the final approximation to the posterior distribution of θ . Representative percentiles of the simulated realizations are shown in figure 10 with comparison to the synthetic field, confirming the potential of the inversion method.

7.5 Example 3: seismic tomography

The propagation of natural or man-made seismic waves through the earth is controlled by certain physical properties of the rock media along the path of the wave propagation. It is conventional to think of the media as constituting a “velocity field”, that is, each location is characterized by a velocity which is determined by relevant rock properties at that location. Suppose seismic P-wave emanates from a source location and is detected at a receiver location, where the “first-arrival traveltimes” of the wave is recorded. The ray

path (i.e. route of the wave propagation) is a complicated function of the velocity field; in particular, it is not a straight line where the velocity field is not uniform. The traveltime is determined by the velocities on the ray path from the source to the receiver. If one knows the velocity field, then both the ray path and the traveltime can be calculated [Vidale, 1988, 1990]. The inverse problem, namely inference of the velocity field given measurements of traveltimes, is called “seismic tomography”. First developed in medical imaging [Arridge, 1999], the tomography technique has seen wide applications in fields such as geophysics [Nolet, 2008] and hydrology [Yeh and Liu, 2000].

We constructed a synthetic tomography problem in two-dimensional space. Define “slowness” s (sm^{-1}) as the reciprocal of velocity. Figure 11 shows a synthetic log-slowness (Y) field (a vertical profile dipping down the ground surface), discretized into a 60×40 grid. Suppose we conducted a tomography experiment with 6 sources and 10 receivers on each vertical boundary, and 15 receivers on the top boundary. Waves from each source were detected by the receivers on the opposite vertical boundary and on the top boundary. The sources and receivers are marked in figure 11. Such a layout of sources and receivers is typical of “crosshole tomography” [Ivansson, 1987, sec. 2.1].

The wave propagation is described approximately by the Eikonal equation [Shearer, 2009, Appendix C]:

$$\left(\frac{\partial t(x_1, x_2)}{\partial x_1}\right)^2 + \left(\frac{\partial t(x_1, x_2)}{\partial x_2}\right)^2 = (s(x_1, x_2))^2, \quad (27)$$

where x_1, x_2 are the spatial coordinates, $t(x_1, x_2)$ is the traveltime from a designated source to (x_1, x_2) , and $s(x_1, x_2)$ is the slowness at location (x_1, x_2) . To illustrate a solution of this system, the contours of first-arrival traveltimes in the synthetic field from a source located on the left boundary are shown in figure 12. This figure clearly illustrates the curved paths of the seismic wave in the heterogeneous media. Note that the wave travels faster in regions of lower slowness (logically enough!).

Solving (27) in the synthetic slowness field and taking logarithmic transform of the resultant traveltimes, we obtained $2 \times 6 \times (10 + 15) = 300$ synthetic traveltime measurements. The solution used the algorithm described by Vidale [1988, 1990].

To reduce the computational cost, we aggregated the synthetic field into a 30×20 grid, that is, a fourth of the resolution of the synthetic field. This introduces *model error*: the synthetic forward model and the forward model used in inversion are not identical—they use different numerical grids. This error is directly analogous to what discretization does to the conceptually continuous field in real-world applications. It may not be trivial to characterize this error quantitatively. In this example we chose to ignore this model error. Although in this particular case the error is small, the notion of ignoring the error is important in illustrating the principles of the anchored inversion methodology. In contrast, some other methods rely on the notion of “error” to construct regularization criteria or likelihood functions (see section 2).

In the iterations of the algorithm, the dimension of the forward data was reduced from 300 to 25–107, thanks to the principal component analysis in step 3 of the algorithm. The algorithm started with 4 anchors, dividing the model domain into 2×2 sub-domains. The parameterization evolved in the 20 iterations and arrived at an anchorset with 16 anchors,

as depicted in figure 13, which once again illustrates the adaptive nature of the algorithm in terms of automatically selecting an anchorset.

Based on the final approximation to the posterior distribution of the model parameter θ , we created 1000 field realizations (on the 30×20 grid). The point-wise median of these realizations is shown in figure 14. Comparison with the synthetic field (aggregated to the 30×20 grid) confirms that the inversion captured important features of the field. The mean of the point-wise standard deviations of the field realizations is 0.015, whereas the mean of the point-wise median absolute deviations (from the synthetic truth) is 0.036. These values should be considered with reference to the range of the synthetic field, which is $(-8.88, -8.53)$.

8 Conclusion

We have proposed a general method for modeling a spatial random field, as a Gaussian process, that is tailored to making use of measurements of nonlinear functionals of the field. These nonlinear functionals are “forward processes (or models)” observed in experimental or natural settings. Inverse of nonlinear forward processes is a major topic with wide scientific applications, as demonstrated in sections 1 and 7.

The method revolves around certain deliberately chosen linear functionals of the field called “anchors”. The anchor parameterization achieves two goals. First, it reduces the dimensionality of the parameter space from that of the spatial field vector to that of the anchor vector (plus possibly a small number of other parameters such as geostatistical parameters). As a result, the dimensionality of the statistical inversion problem is separate from the spatial resolution of the forward model. Second, the parameterization establishes a statistical connection between the model parameter (i.e. anchors) and the data (i.e. forward process outcome). This statistical connection is known in mechanism and can be studied through simulations. The existence of this connection makes the inversion task a statistical one, whether or not the data contain model and measurement errors. This makes the method a fundamental departure from some existing approaches.

Although the framework has a Bayesian flavor, it does not go down a typical Bayesian computational path, because the likelihood function is unknown. The computational core lies in an iterative algorithm (section 4) centered on kernel density estimation. The algorithm is general and flexible in several aspects. First, it does not require a statistical analysis of the parameter-data connection, but only needs the ability to evaluate the forward model (usually by running a numerical code), which is treated as a black box operation. Although this study has focused on deterministic forward models, it is possible to accommodate stochastic ones. Second, the algorithm is flexible with regard to the prior specification. Third, the algorithm is able to accommodate flexible forms of errors (section 4.2). Fourth, different forward models (hence forward data) could be used in different iterations. The last point, not elaborated in this article, makes it possible to assimilate multiple datasets sequentially or in other flexible ways. These features make the algorithm a general procedure for approximating the posterior distribution where the likelihood is unknown but the model (which corresponds to the likelihood) can be simulated.

A critical component of the method concerns selecting the anchor parameterization in an adaptive, automatic fashion as iteration proceeds (section 5). The resultant anchor config-

uration is tailored to features of the field Y and the computational effort already invested. For example, if more iterations are performed, the algorithm in general will introduce more anchors to achieve more detailed representation of the field.

We used several synthetic examples to demonstrate a number of features of the methodology, including the capabilities of using linear data (example 1), using highly correlated data (all examples, especially examples 2 and 3), assimilating multiple datasets of different physical natures (example 2), and ignoring model or measurement errors (example 3). In view of the dimensionality of the parameter vector (in the tens) and that of the data (tens or hundreds), the algorithm showed potential in efficiency.

We have implemented the method in a R package named `AnchoredInversion`. Currently the package uses the packages `RandomFields` [Schlather, 2001] for simulating random fields and `MASS` (its function `mvrnorm`) [Venables and Ripley, 2002] for generating multivariate normal random numbers. The graphics in this article were created using the package `lattice` [Sarkar, 2008].

Acknowledgement: The work presented in this article was in part supported by the Excellent State Key Lab Fund no. 50823005, National Natural Science Foundation of China, and the R&D Special Fund for Public Welfare Industry no. 201001080, Chinese Ministry of Water Resources. The NED data (used in this study as synthetic data for the examples) from the National Map Seamless Server are open to the public, and are available from U.S. Geological Survey, EROS Data Center, Sioux Falls, South Dakota. Professor Paul Switzer suggested presenting the geostatistical parameterization as an extension (instead of being integrated from the beginning). The author thanks an anonymous reviewer for constructive comments.

References

- M. Aitkin. Posterior Bayes factors. *J. R. Stat. Soc., B*, 53(1):111–142, 1991. URL <http://www.jstor.org/stable/2345730>.
- S. R. Arridge. Optical tomography in medical imaging. *Inverse Problems*, 15:R41–R93, 1999. doi: doi:10.1088/0266-5611/15/2/022.
- J. Besag, J. York, and A. Mollié. Bayesian image restoration, with two applications in spatial statistics. *Ann. Inst. Statist. Math.*, 43(1):1–59, 1991.
- K. Beven. Prophecy, reality and uncertainty in distributed hydrological modelling. *Adv. Water Resour.*, 16:41–51, 1993.
- K. P. Bube and R. Burridge. The one-dimensional inverse problem of reflection seismology. *SIAM Review*, 25(4):497–559, 1983.
- J. Doherty. Ground water model calibration using pilot points and regularization. *Ground Water*, 41(2):170–177, 2003.
- M. H. H. W. Engl and A. Neubauer. *Regularization of Inverse Problems*. Kluwer, 2000.
- E. I. George. Bayesian model selection. In S. Kotz, C. B. Read, N. Balakrishnan, and B. Vidakovic, editors, *Encyclopedia of Statistical Sciences*. John Wiley & Sons, Inc., 2006.
- J. Geweke. Bayesian inference in econometric models using Monte Carlo integration. *Econometrica*, 57(6):1317–1339, 1989.

- T. R. Ginn and J. H. Cushman. Inverse methods for subsurface flow: a critical review of stochastic techniques. *Stoch. Hydrol. Hydraul.*, 4:1–26, 1990.
- D. Higdon. Space and space-time modeling using process convolutions. In *Quantitative Methods for Current Environmental Issues*, pages 37–56. Springer, 2002.
- D. Higdon, H. Lee, and C. Holloman. Markov chain Monte Carlo-based approaches for inference in computationally intensive inverse problems. In *Bayesian Statistics*, volume 7. Oxford University Press, 2003.
- J.-K. Huang and K. T. Lee. Influences of spatially heterogeneous roughness on flow hydrographs. *Adv. Water Resour.*, 32:1580–1587, 2009.
- S. Ivansson. Crosshole transmission tomography. In G. Nolet, editor, *Seismic Tomography*, pages 159–188. D. Reidel Publishing Company, 1987.
- J. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*. Applied Mathematical Sciences 160. Springer, 2005.
- R. E. Kass and A. E. Raftery. Bayes factors. *J. Am. Stat. Assoc.*, 90(430):773–795, 1995.
- A. Kirsch. *An Introduction to the Mathematical Theory of Inverse Problems*. Springer, 1996.
- C. Kravaris and J. H. Seinfeld. Identification of parameters in distributed parameter systems by regularization. *SIAM J. Control Optim.*, 23(2):217–241, 1985.
- H. K. H. Lee, D. M. Higdon, Z. Bi, M. A. R. Ferreira, and M. West. Markov random field models for high-dimensional parameters in simultaitons of fluid flow in porous media. *Technometrics*, 44(3):230–241, 2002. doi: 10.1198/004017002188618419.
- P. Marjoram, J. Molitor, V. Plagnol, and S. Tavaré. Markov chain Monte Carlo without likelihoods. *Proceedings of the National Academy of Sciences*, 100:15324–15328, 2003. doi: 10.1073/pnas.0306899100.
- J. S. Marron and M. P. Wand. Exact mean integrated squared error. *Ann. Statist.*, 20(2): 712–736, 1992.
- G. McLachlan. *Finite Mixture Models*. John Wiley & Sons, Inc., 2000.
- S. Nakagiri. Review of Japanese work of the last ten years on identifiability in distributed parameter systems. *Inverse Problems*, 9:143–191, 1993.
- G. N. Newsam and I. G. Enting. Inverse problems in atmospheric constituent studies: I. determination of surface sources under a diffusive transport approximation. *Inverse Problems*, 4:1037–1054, 1988.
- G. Nolet. *A Breviary of Seismic Tomography*. Cambridge University Press, 2008.
- F. O’Sullivan. A statistical perspective on ill-posed inverse problems. *Statistical Science*, 1(4):502–518, 1986.
- A. Owen and Y. Zhou. Safe and effective importance sampling. *J. Am. Stat. Assoc.*, 95(449):135–143, 2000.
- M. Sambridge and K. Mosegaard. Monte Carlo methods in geophysical inverse problems. *Reviews of Geophysics*, 40(3), 2002. doi: 10.1029/2000RG000089.
- D. Sarkar. *Lattice: Multivariate Data Visualization with R*. Springer, 2008.
- J. A. Scales and L. Tenorio. Prior information and uncertainty in inverse problems. *Geophysics*, 66(2):389–397, 2001.

- M. Schlather. Simulation and analysis of random fields. *R News*, 1(2):18–20, June 2001.
- G. Schoups, J. A. Vrugt, F. Fenicia, and N. C. van de Giesen. Corruption of accuracy and efficiency of Markov chain Monte Carlo simulation by inaccurate numerical implementation of conceptual hydrologic models. *Water Resour. Res.*, 46:W10530, 2010. doi: 10.1029/2009WR008648.
- F. W. Schwartz and H. Zhang. *Fundamentals of Ground Water*. John Wiley & Sons, Inc., 2003.
- P. M. Shearer. *Introduction to Seismology*. Cambridge University Press, 2nd edition, 2009.
- M. Short, D. Higdon, L. Guadagnini, A. Guadagnini, and D. M. Tartakovsky. Predicting vertical connectivity within an aquifer system. *Bayesian Analysis*, 5(3):557–582, 2010.
- M. L. Stein. *Interpolation of Spatial Data: Some Theory for Kriging*. Springer, 1999.
- M. A. Tanner. *Tools for Statistical Inference*. Springer, 3rd edition, 1996.
- A. Tarantola. *Inverse Problem Theory and Methods for Model Parameter Estimation*. SIAM, Philadelphia, 2005.
- A. Tarantola and B. Valette. Inverse problems = question for information. *Journal of Geophysics*, 50:159–170, 1982.
- L. Tenorio. Statistical regularization of inverse problems. *SIAM Review*, 43(2):347–366, 2001.
- A. Tikhonov and V. Arsenin. *Solution of Ill-posed Problems*. Winston, Washington, DC, 1977.
- P. J. van Leeuwen and G. Evensen. Data assimilation and inverse methods in terms of a probabilistic formulation. *Mon. Weather Rev.*, 124:2898–2913, 1996.
- W. N. Venables and B. D. Ripley. *Modern Applied Statistics with S*. Springer, 4th edition, 2002.
- J. Vidale. Finite-difference calculation of travel times. *Bulletin of the Seismological Society of America*, 78(6):2062–2076, 1988.
- J. E. Vidale. Finite-difference calculation of traveltimes in three dimensions. *Geophysics*, 55(5):521–526, 1990.
- B. E. Vieux. *Distributed hydrologic modeling using GIS*. Kluwer, 2nd edition, 2004.
- G. Welch and G. Bishop. An introduction to the Kalman filter. Technical Report TR 95-041, Department of Computer Science, University of North Carolina at Chapel Hill, 1995. Updated in 2006.
- M. West. Approximating posterior distributions by mixture. *J. R. Stat. Soc., B*, 55(2):409–422, 1993.
- T.-C. J. Yeh and S. Liu. Hydraulic tomography: development of a new aquifer test method. *Water Resour. Res.*, 36(8):2095–2105, 2000.
- W. W.-G. Yeh. Review of parameter identification procedures in groundwater hydrology: the inverse problem. *Water Resour. Res.*, 22(2):95–108, 1986.
- Z. Zhang. Iterative posterior inference for Bayesian kriging. *Stoch. Environ. Res. Risk Assess.*, 26(7):913–923, 2012. doi: 10.1007/s00477-011-0544-y.
- Z. Zhang, D. Beletsky, D. J. Schwab, and M. L. Stein. Assimilation of current measurements into a circulation model of Lake Michigan. *Water Resour. Res.*, 43:W11407, 2007. doi: 10.1029/2006WR005818.

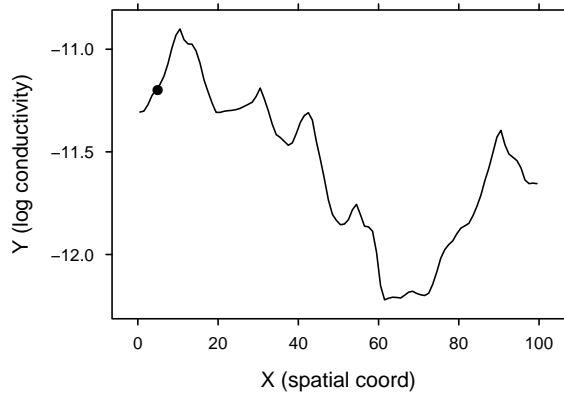


Figure 1: Synthetic field of log hydraulic conductivity (meter/second) for example 1 in section 7. The dot on the curve indicates the single direct measurement (acting as available linear data).

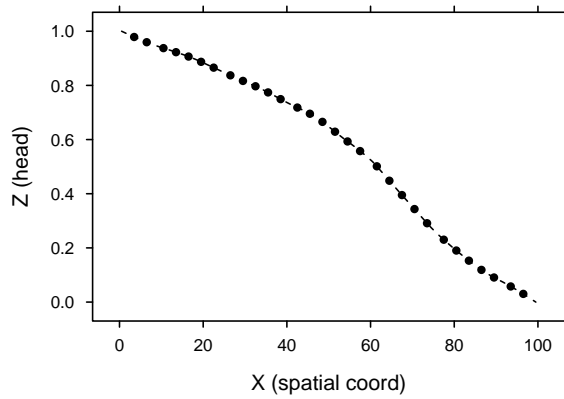


Figure 2: Steady-state head level (m) in the synthetic log-conductivity field, under specified boundary conditions, of example 1 in section 7. The dots indicate synthetic head measurements.

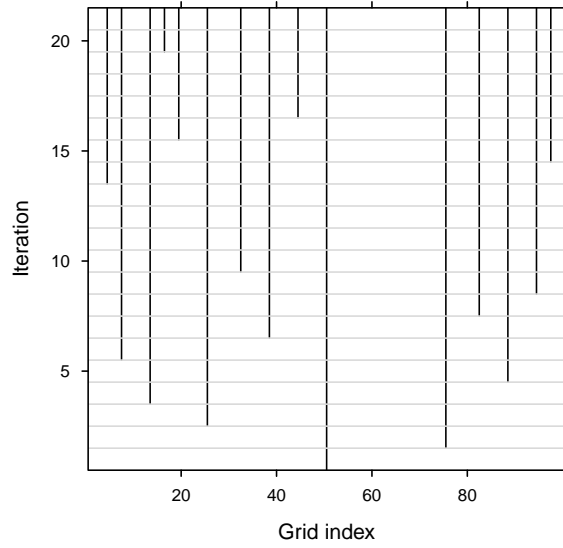


Figure 3: Evolution of anchorsets in the iterative algorithm for example 1 in section 7. In each iteration, the model domain is divided by the vertical bars; the average value of Y in each division constitutes an anchor.

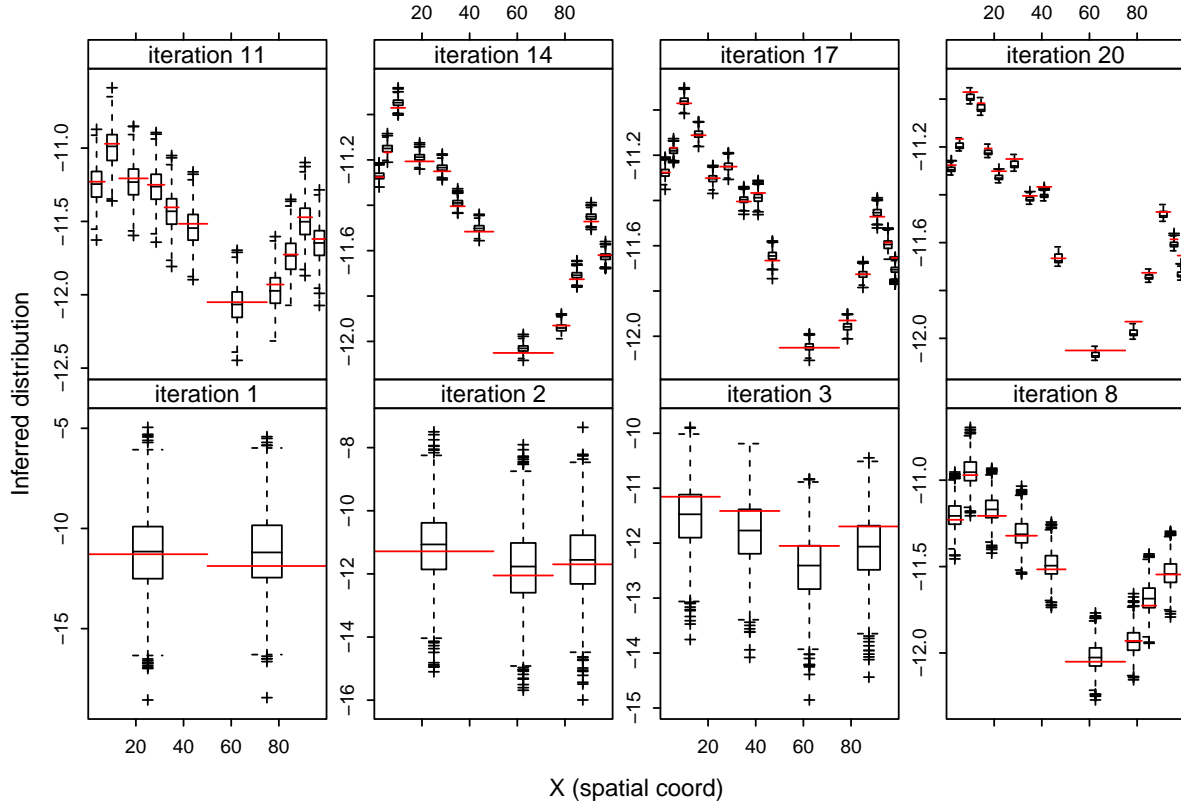


Figure 4: Marginal distribution of each anchor, represented by a boxplot, in the iterative algorithm for example 1 in section 7. Each box is located horizontally at the center of the “support” of its corresponding anchor. The red lines indicate the “target value” of each anchor (obtained using the synthetic true field).

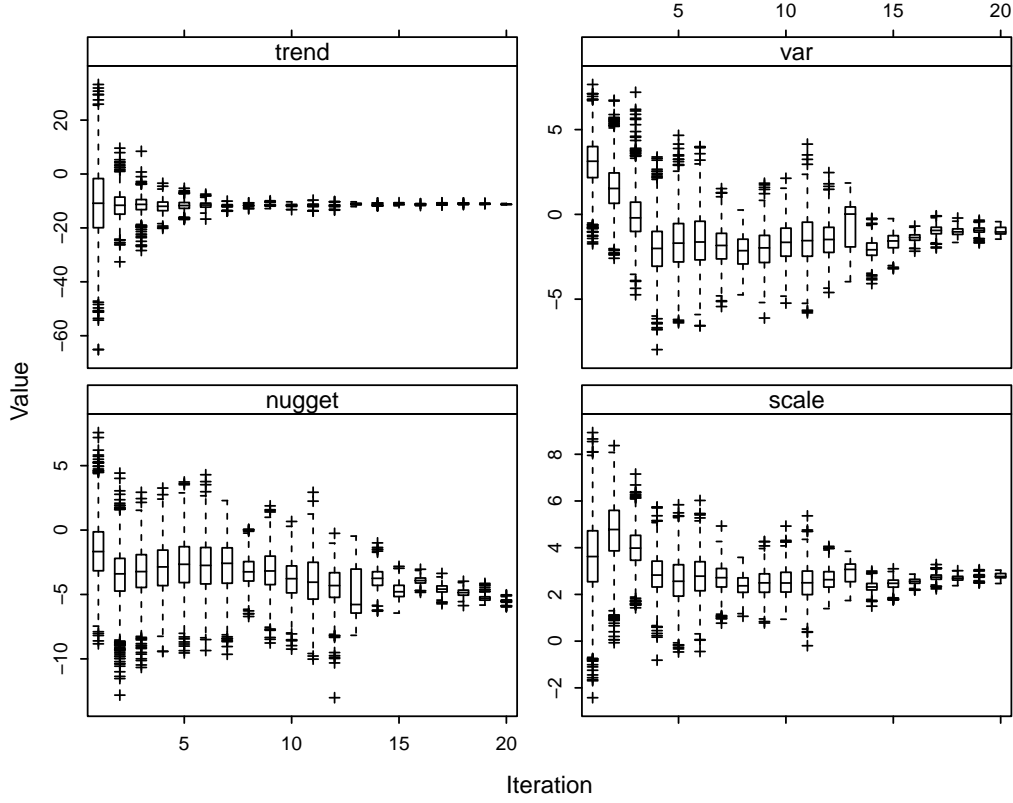


Figure 5: Evolution of the estimated marginal posterior distributions of each geostatistical parameter in the iterative algorithm for example 1 in section 7. The distributions are represented by boxplots of the samples obtained in step 1 of the algorithm.

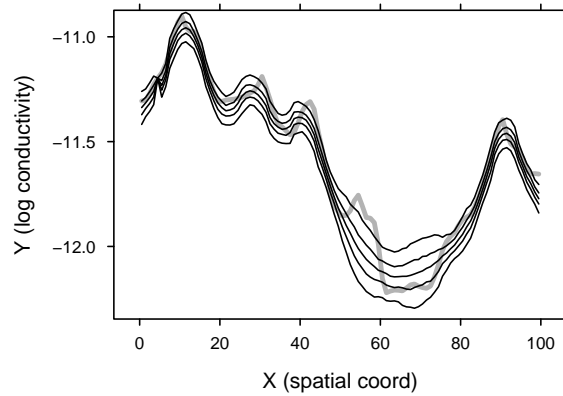


Figure 6: Pointwise 5th, 25th, 50th, 75th, and 95th percentiles of 1000 field simulations based on the anchorset and approximate posterior distribution in the final iteration of example 1 in section 7. The thick gray curve is the synthetic field. (Note how the direct measurement was reproduced exactly.)

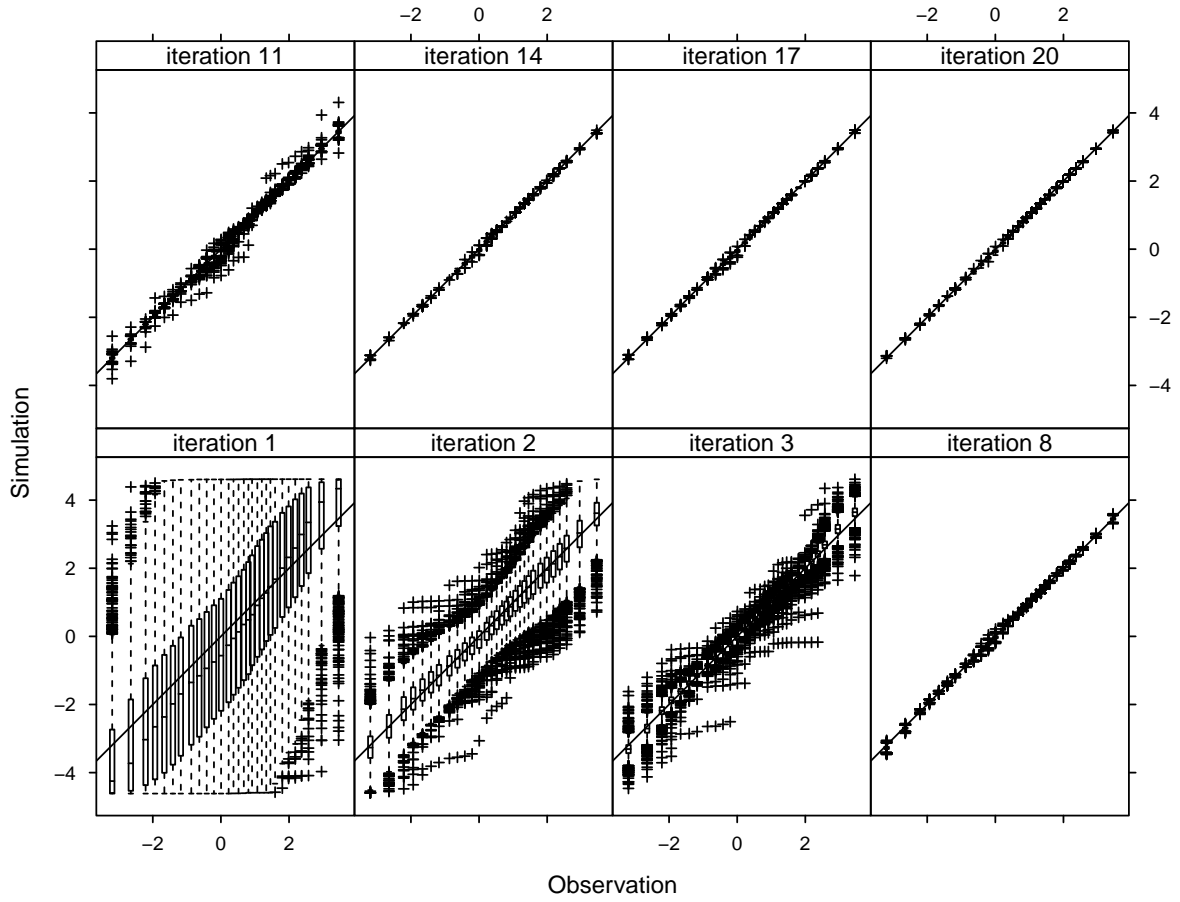


Figure 7: Boxplot of each component of the forward predictions (z_i) sampled in step 2 of selected iterations in example 1 of section 7. The horizontal location of each boxplot is at the observed value of the corresponding forward datum. The straight reference lines are $y = x$.

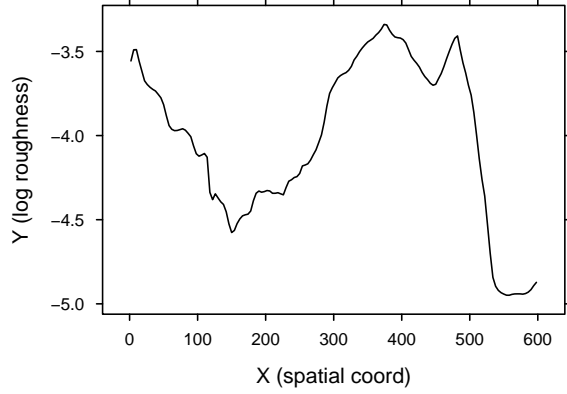


Figure 8: Synthetic field of log roughness ($\text{sm}^{-1/3}$) for example 2 in section 7.

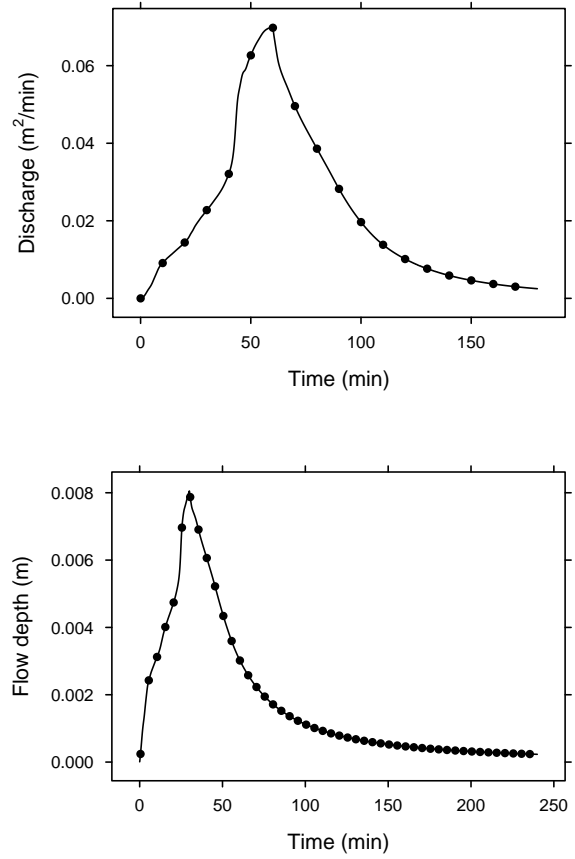


Figure 9: Discharge (top; in the first rainfall and measurement scenario) and flowdepth (bottom; in the second rainfall and measurement scenario) processes at the downstream boundary in example 2 of section 7. The dots indicate synthetic measurements.

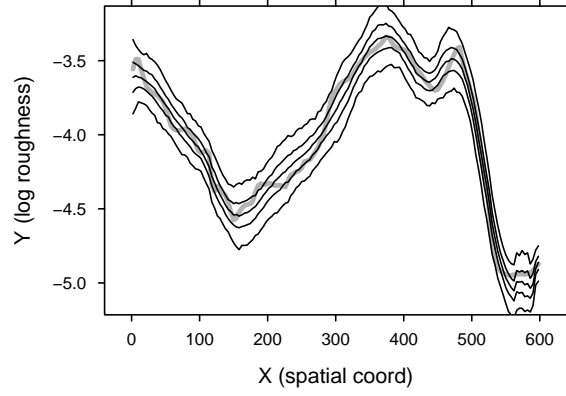


Figure 10: Pointwise percentiles of 1000 field simulations in example 2 of section 7. See the caption of figure 6 for explanations.

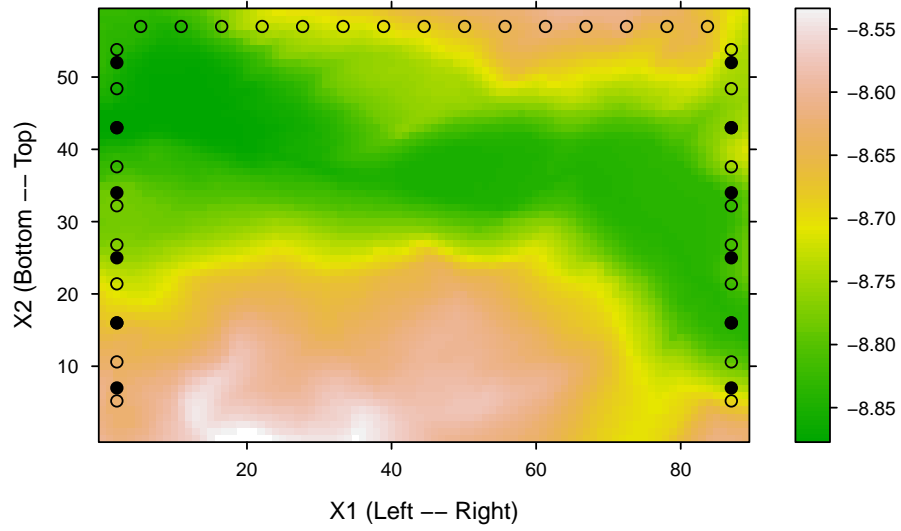


Figure 11: Synthetic field of log slowness (second/meter) in example 3 of section 7. The dots indicate wave sources; the circles indicate receivers.

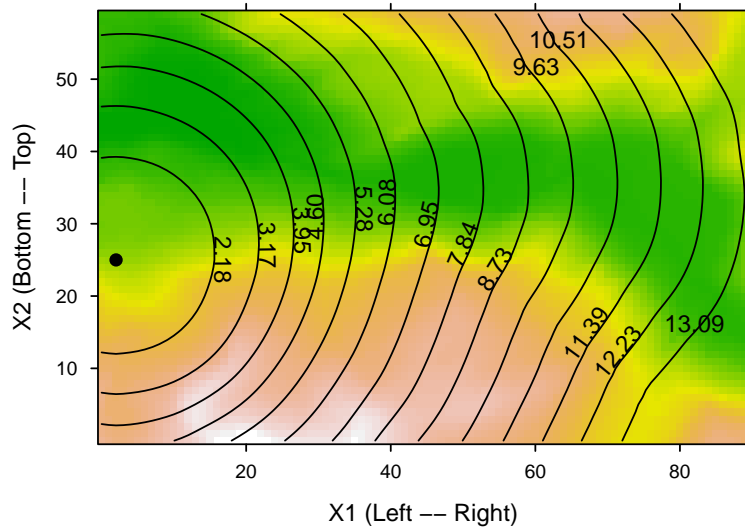


Figure 12: First-arrival times (10^{-3} s) of seismic wave, originating from the source indicated by the dot on the left, in the synthetic slowness field (on 60×40 grid) of example 3 in section 7.

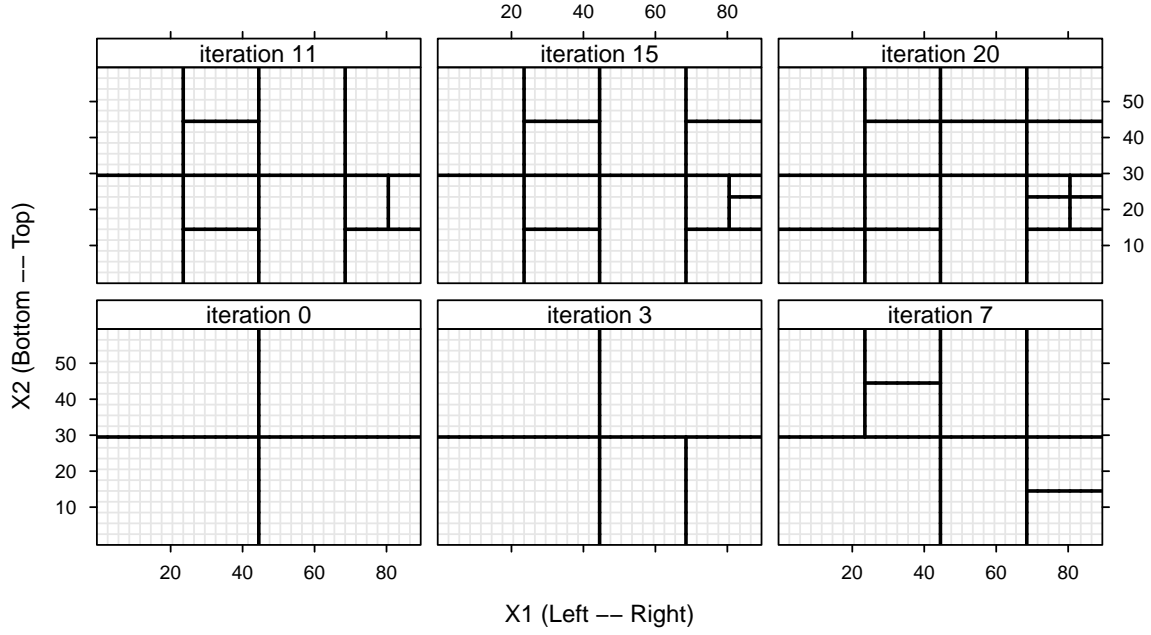


Figure 13: Layout of anchorsets in selected iterations of example 3 in section 7. The model domain is divided into sub-domains; the average value of Y in each sub-domain constitutes an anchor.

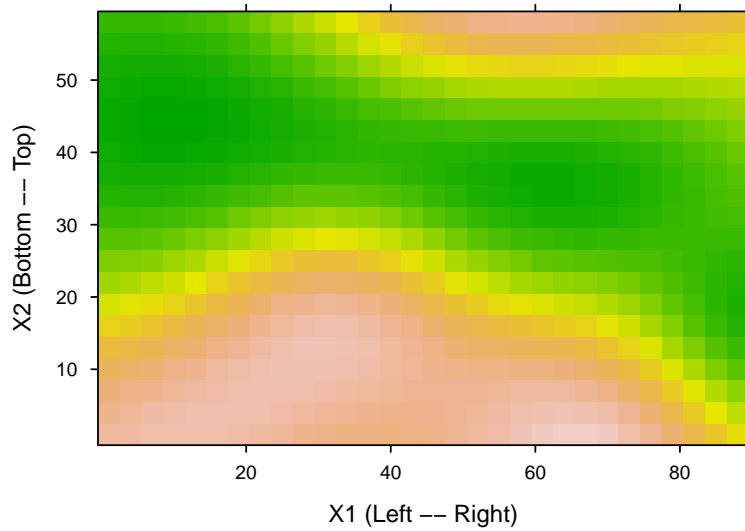


Figure 14: Pointwise median of 1000 field simulations based on the anchorset and approximate posterior distribution in the final iteration of example 3 in section 7.